

004  
B81

I.M. BOYNAZAROV



# Linux<sup>TM</sup>

## OPERATSION

### TIZIMI

O'ZBEKISTON RESPUBLIKASI AXBOROT  
TEKNOLOGIYALARI VA KOMMUNIKATSIYALARINI  
RIVOJLANTIRISH VAZIRLIGI

MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT  
AXBOROT TEKNOLOGIYALARI UNIVERSITETI

I.M. BOYNAZAROV

# LINUX OPERATION TIZIMI

(O'quv qo'llanma)

TOSHKENT - 2021

**UO'K: 004.451(075)**

**KBK: 32.81**

**B 81**

**Boynazarov I.M. Linux operatsion tizimi. (O'quv qo'llanma). –T.: «Nihol print» OK, 2021. – 320 b.**

**ISBN 978–9943–7028–9–9**

O'quv qo'llanmada Linux operatsion tizimining zamonaviy distributivlarida yadro skriptlarini yozish, skriptlarni bajariluvchi faylga aylantirish, tarmoq DNS serveri, WEB serverlar, FTP server va Elektron pochta serverlarini sozlash masalalari ko'rib chiqilgan. Har bir serverni sozlash uchun zarur bo'ladigan konfiguratsiya fayllarini tahrirlash, ularni saqlash, qayta yuklash bo'yicha amaliy ko'nikmalarni shakllantirish uchun misollar va laboratoriya mashg'ulotlari keltirilgan.

O'quv qo'llanma 5330500 – Kompyuter injiniringi, 5330600 – Dasturiy injiniringi bakalavr ta'lim yo'nalishlari talabalari uchun mo'ljallangan.

**UO'K: 004.451(075)**


**KBK: 32.81**

### **Taqrizchilar:**

**N.Zaynalov** – Muhammad al-Xorazmiy nomidagi TATU Samarqand filiali “Axborot xavfsizligi” kafedrasi mudiri, dotsent.

**E.Urinboev** – Samarqand davlat universiteti “Matematik modellashtirish va kompleks dasturlashtirish” kafedrasi dotsenti.

 - e'tibor qilish zarur bo'lgan tushunchalar.

 - savol va topshiriqlar.

**ISBN 978–9943–7028–9–9**

**© «Nihol print» OK nashriyoti, 2021.**

## KIRISH

Mazkur o'quv qo'llanma Linux operatsion tizimi yadrosi darajasida scriptlar yozish, ularni saqlash, ishga tushirish, bajariluvchi faylga aylantirish masalalari hamda Linux-da administratsiyalash masalasiga oid asosiy bilimlar to'liq yoritib o'tilgan.

Qo'llanmada Shell script asoslari, Linux-ning ishga tushirish (yuklovchi) tizimlari, qurilmalar va log fayllari bilan ishlash, DNS serveri, WEB serverlar, FTP server va Elektron pochta serverlarini sozlash kabi amaliy masalalar misollar yordamida tushuntirib berilgan. Talabalar mustaqil ishlash ko'nikmalarini shakllantirish maqsadida nazorat savollari ham laboratoriya ishlari keltirilgan.

Qo'llanma ikki qismdan tashkil topgan. Birinchi qismi Linux operatsion tizimida administratsiyalash masalalariga bag'ishlangan bo'lib, unda nazariy ma'lumotlar bilan birgalikda amaliy masalalar keltirilgan. Ikkinchi qismda birinchi qismda o'rganilgan bilimlarni mustahkamlash uchun laboratoriya ishlari berilgan. Bu laboratoriya ishlari guruh bo'lib, ishlashga mo'ljallangan bo'lib, har bir ishda kompyuterlar tarmog'ida bajariladigan bir nechta topshiriqlar amaliy ko'rsatmalar bilan taqdim etilgan.

O'quv qo'llanmani tayyorlashda misol sifatida keltirilgan barcha amaliy masalalar Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universitetining Samarqand filialidagi kompyuterlarda amaliyotga tatbiq etilib, olingan natijalarga asoslangan fikrlar bayon qilingan. Kompyuterlar tarmog'ining serverlarini tiklashda filialdagi [www.zukko.uz](http://www.zukko.uz) domeni joylashgan serverdan foydalanildi.

O'quv qo'llanma 5330500 – Kompyuter injiniringi, 5330600 – Dasturiy injiniringi bakalavr ta'lim yo'nalishlari talabalari uchun mo'ljallangan.

## 1-BOB. SHELL SCRIPT ASOSLARI

### 1.1-§. Shell script – asosiy tushunchalar

Shellda *shell script* deb ataluvchi oʻrnatilgan dasturlash tili mavjud. Yaʼni, *shell* qobigʻiga oʻrnatilgan ikkita asosiy interpretator va ikki xil dasturlash tillari *Bourne shell* va *C shell* mavjud.

*C shell* tili oʻzining sitaksisi boʻyicha C dasturlash tilini eslatadi. Aynan shuning uchun ham bu qobiq shunday nomlangan.

*Bourne shell* tili oʻzining sintaksisi va xususiyatlariga koʻra mustaqil til hisoblanadi. Nima uchun Linux administratori *shell script*-ni bilishi kerak? Gap shundaki, bu tillar administratorning yordamchi instrumenti sifatida ishlab chiqilgan.

UNIX ning muhim xususiyatlaridan biri – bu har biri oʻz vazifasini bajaruvchi kichik utilitlarning katta toʻplami mavjudligidir. Komandalarni konveyerli qayta ishlash yordamida murakkab masalalarni bajarish uchun ushbu utilitlarni birlashtirish mumkin. Afsuski, konveyerli qayta ishlash bilan barcha masalalarni hal qilish mumkin emas, baʼzida koʻproq narsa talab etiladi. Dasturni ixtiyoriy dasturlash tilida yozish mumkin, lekin ushbu dasturlash tilini oʻrganish kerak, tizimda esa uning kompilyatori oʻrnatilgan boʻlishi shart.

Shell script-da dasturlarni kompilyatsiya qilish kerak emas, bunda dasturlar qobiqqa oʻrnatilgan interpretator orqali bajariladi. Yaʼni, shell script – bu interpretatsiya qilinadigan til hisoblanadi.

Bundan tashqari barcha Linux tizimlari shell script tilidagi bir qancha fayllardan tuzilgan, Linux administratori ushbu fayllarni oʻz ixtiyori bilan oʻzgartirishi mumkin, buning uchun esa dasturlash tili boʻyicha tayanch bilimlarga ega boʻlishi kerak.

Shell script-dagi dastur fayli – bu foydalanuvchilar tomonidan komandalar satrida kiritilishi mumkin boʻlgan komandalar toʻplami hisoblanadi. Agar fayl bajarishga yuklansa, ushbu faylda yozilgan barcha komandalar bajariladi. Lekin, shell-dagi buyruqlarning oddiy bajarilishi bilan bir qatorda, koʻplab dasturlash tillariga xos boʻlgan qoʻshimcha imkoniyatlari ham mavjud – bular oʻzgaruvchilar, massivlar, shartli operatorlar, takrorlash operatorlari va boshqalarni qoʻllash imkoniyatlaridir.

Ushbu bobda Bourne shell-ga oʻrnatilgan asosiy tilni oʻrganamiz, yaʼni Linux-ni ishga tushirish tizimidagi fayllarda qoʻllaniladigan shakllarini oʻrganib chiqamiz.

## 1.2-§. Bourne shell ilovasini yuklash

Shell script fayllari ishga tushishi uchun uni bajariluvchi binar faylga aylantirish zarur. Faylning birinchi satrida dastur bajarilishi uchun chaqiriladigan interpretatorga yo‘l ko‘rsatilishi kerak. Ya’ni, Linux-da eng ko‘p ishlatiladigan interpretator Bourne shell-ning turi – bash uchun faylning birinchi satrida quyidagi satrni yozilish kerak:

```
#! /bin/bash
```

Shell script-da yozilgan dasturni yuklashning yana bir varianti mavjud bo‘lib, bu fayl dastur bilan ishga tushirilmaydi. Bunday holatda, fayl aynan biz foydalanayotgan shell komandalari bilan ishlashiga ishonch hosil qilishimiz kerak. Bu dasturlarni bajarilishi uchun yuklashda komandalar satriga quyidagi komandani terish kerak:

```
.program!
```

Shundan keyin ko‘rsatilgan dastur ayni vaqtda ishlab turgan qobiqning nusxasi (namunasi) sifatida bajariladi. Ya’ni, dasturning bajarilishi uchun yangi jarayon (process) hosil bo‘lmaydi. Agar dastur kodida **exit** operatori bajarilsa, tizimdan chiqib ketadi.

Xuddi shunday interpretatorni komandalar satriga ochiqchasiga (oshkor holatda) chaqirish mumkin:

```
bash program.sh
```

## 1.3-§. O‘zgaruvchilar



*/root/bin katologida o‘zgaruvchilardan foydalanish bo‘yicha misollar keltirilgan fayllar mavjud bo‘lib, ushbu fayllar asosida shell script-da dasturlash asoslari o‘rganiladi.*

Shell script o‘zgaruvchilarni qo‘llash mumkin bo‘lgan eng qulay dasturlash tili hisoblanadi. Shell script o‘zgaruvchilari – interpretator (bash) dasturlash muhitining o‘zgaruvchilaridir.

Birinchi o‘rinda o‘zgaruvchilarning tiplashtirilmaganligini alohida ta’kidlab o‘tish kerak. O‘zgaruvchilarning barcha qiymatlari satr sifatida o‘qiladi. Faqat o‘zgaruvchilar matematik ifodalarda qo‘llanilsagina uning turi tekshiriladi.

Shell script-da o‘zgaruvchilarning «ta’sir sohasi» tushunchasi mavjud emas. O‘zgaruvchilarga dastur kodining ixtiyoriy joyidan, jumladan funksiyalar (qism dasturlar)da ham murojaat qilish mumkin. Agar o‘zgaruvchi funksiyada aniqlangan bo‘lsa ham, unga global

o'zgaruvchi sifatida dastur kodining ixtiyoriy joyida murojaat qilish mumkin bo'ladi.



*Bashning 2-versiyasida **local** operatori yordamida lokal o'zgaruvchilarni aniqlash imkoni paydo bo'ldi. Lekin bu imkoniyat **Bourne shell**-ning boshqa versiyalari uchun standart sifatida qabul qilinmagan.*

Agar aniqlanmagan o'zgaruvchiga murojaat tashkil etilsa, interpretator xatolik haqida xabar bermaydi. Oddiygina bo'sh satr qoldiradi xolos.

Quyida o'zgaruvchilarning qo'llanishiga doir dastur matni keltirilgan.

```
1 #!/bin/bash
2 CAR= "CAPTIVA - eng yaxshi kompyuter"
3 echo "CAR: "
4 echo CAR
5 echo '$CAR: '
6 echo $CAR
```

Dastur kodining 1-satrida ushbu dasturning bajarilishi uchun zarur bo'lgan interpretator turi keltirilgan.

O'zgaruvchining aniqlash uchun uning nomini yozish kerak (katta va kichik harflar farqlanadi), so'ngra "tenglik" belgisini qo'yib o'zgaruvchining qiymati yoziladi (2-satr). Agar o'zgaruvchi qiymatida probel bo'lsa, u holda qiymat bittalik yoki ikkitalik qo'shtirnoq ichiga olib yozilishi kerak.

```
CAR= "CAPTIVA - eng yaxshi kompyuter"
```

O'zgaruvchining qiymatini olish uchun uning nomi oldiga \$ belgisi qo'shib yoziladi (6-satr).

```
echo $CAR
```

3-satrd ekranga - **CAR:** yozuvi chiqariladi;

4-satrd ekranga – **CAR** yozuvi chiqariladi;

5-satrd bittalik qo'shtirnoq ishlatilgan. Shell script-da bu maxsus belgi hisoblanadi – bittalik qo'shtirnoq ichidagi barcha belgilar ekranga to'liq chiqariladi. Ya'ni bu yerdagi \$ belgisi maxsus belgi sifatida inetrpretatsiya qilinmaydi, natijada ekranga **\$CAR:** yozuvi chiqariladi.

## 1.4-§. Massiv o'zgaruvchilari

Shell script-da massiv o'zgaruvchisini qo'llash mumkin. Klassik bourne shell interpretorida bitta cheklov mavjud bo'lib, massiv elementlari soni 1024 tadan ko'p bo'lmasligi kerak. Bashda esa bu cheklov olib tashlangan.

Quyida dastur faylining mazmuni nomerlangan satrlarda berilgan.

```
1 #!/bin/bash
2 #Massiv
3 #CAR[0]=CAPTIVA
4 #CAR[1]=NEXIA
5 #CAR[2]=LACETTI
6 #CAR[5]=SPARK
7 #CAR[10]=DAMAS
8 #CAR=( [0]=CAPTIVA [1]=NEXIA [2]=LACETTI
[5]=SPARK [10]=DAMAS)
9 CAR=(CAPTIVA NEXIA LACETTI SPARK DAMAS)
10 echo "*****"
11 echo "CAR[0]=${CAR[0]}"
12 echo "CAR[1]=${CAR[1]}"
13 echo "CAR[2]=${CAR[2]}"
14 echo "CAR[3]=${CAR[3]}"
15 echo "CAR[4]=${CAR[4]}"
16 echo "*****"
17 echo "ALL - ${CAR[*]}"
18 echo "UNO - ${CAR[@]}"
```

Bash-da massivni aniqlashning bir necha usullari mavjud. Misolda ko'rsatilgan faylda dastlabki ikkita usuli izohlangan (satr boshida # belgisi ishlatilgan – bu izoh kiritishni bildiradi).

Dastlab misolda (3-satrdan 7-satrgacha) elementlar massiv nomini ko'rsatish orqali aniqlangan. Keyin kvadrat qavsda massivdagi element nomeri va uning qiymatlari oddiy o'zgaruvchilar kabi tenglik orqali berilgan, masalan:

```
CAR[1]= NEXIA
```

Massiv elementlari tartib bilan aniqlanmaydi: 0, 1, 2, 5 va 10. Diqqat bilan e'tibor bersak, massivning mavjud bo'lmagan



elementlariga murojaatni interpretator hech qanday xatolik haqida xabar bermasdan qabul qiladi.

Massivni ikkinchi usulda aniqlash 8-satrdan berilgan.

```
#CAR=( [0]=CAPTIVA [1]=NEXIA [2]=LACETTI  
[5]=SPARK [10]=DAMAS)
```

Bu misolda dastlab massivning nomi yozilgan, keyin tenglik orqali massiv elementlarining qiymatlari qavs ichida berilgan.

Massivni aniqlashga uchinchi misol (9-satr) oldingi misoldagiga o'xshash bo'lib, bunda elementlar nomeri ko'rsatilmaydi. Bunda qiymatlar berilgan tartibi bo'yicha o'zlashtiriladi, ya'ni dastlab nolinch element, keyin ikkinchi va h.k.

```
CAR=(CAPTIVA NEXIA LACETTI SPARK DAMAS) .
```

Massiv elementi qiymatini olish uchun quyidagi konstruktsiya qo'llaniladi:

```
#{CAR[0]}
```



*Figurali qavsga e'tibor berish kerak. Massivlar bilan ishlashda bu alohida ahamiyatga ega.*

Misolda massivning barcha elementlarini chiqarish 17- va 18-satrlarda berilgan. Buning uchun massiv elementi nomeri o'rniga @ belgisi yoki \* belgisi ishlatilgan.

```
#{CAR[*]}
```

```
#{CAR[@]}
```

## 1.5-§. Tashqi o'zgaruvchilar

Shell script tilida yozilgan dasturda tashqi o'zgaruvchilarga kirish (murojaat qilish) imkoni mavjud bo'lib, ularning qiymatlarini o'zgartirish, shuningdek yangi tashqi o'zgaruvchilarni aniqlash mumkin, misol uchun:

```
1 #!/bin/bash
```

```
2 #PATH=$PATH:~/bin; export PATH
```

```
3 #export PATH=$PATH:~/bin
```

```
4 echo "Workdir=$PWD"
```

```
5 echo "Bu UID=$UID"
```

```
6 echo "Bash level=$SHLVL"
```

## 7 echo "Random=\$RANDOM"

Klassik Bourne shell-da yangi tashqi o'zgaruvchilarni hosil qilish uchun, dastlab qobiq o'zgaruvchilari yaratiladi, keyin esa u eksport qilinadi.

O'zgaruvchilarni eksport qilish uchun **export** operatori qo'llaniladi (2-satr).

```
PATH=$PATH:~/bin
export PATH
```

Bashda o'zgaruvchilarni aniqlash va uni eksport qilish bir vaqtda bajarilishi mumkin (3-satr).

```
export PATH=$PATH:~/bin
```

Tashqi o'zgaruvchilarga ushbu jarayon hamda shu dastur orqali yaratilgan boshqa barcha jarayonlardan murojaat qilish mumkin bo'ladi. Tizimning boshqa jarayonlariga bu o'zgaruvchilar ko'rinmaydi.

Barcha tashqi o'zgaruvchilar va funksiyalarni ko'rish uchun **set** operatoridan foydalanish mumkin. **Env** dasturi faqat o'zgaruvchilarni ko'rsatadi, **export** esa –faqat eksport qilishga tayyorlangan o'zgaruvchilarni ko'rsatadi.

O'zgaruvchilarni o'chirish **unset** operatori orqali bajariladi. Masalan:

```
unset CAR
```

Shell-da juda ko'p o'rnatilgan o'zgaruvchilar mavjud. Jadvalda ulardan ba'zilar keltirib o'tilgan. Barcha o'rnatilgan o'zgaruvchilar tavsifini ko'rish uchun interpretatorning ma'lumotnoma yo'riqnomasiga kirish kerak<sup>1</sup>.

O'zgaruvchilar	Tavsifi
NOME	Foydalanuvchining ildiz katologi
LOGNAME	Foydalanuvchining registratsiya nomi
MAILCHECK	Foydalanuvchining elektron pochtasidagi yangi xatlarni tekshirish uchun sarflangan sekundlar soni

<sup>1</sup> man bash - buyrug'i yordamida

PATH	Agar foydalanuvchi oxirgi yuklash vaqtida aniq bir yo'lni ko'rsatmagan bo'lsa, interpretator dasturni izlash uchun qo'llaniladigan, o'zaro ikki nuqta bilan ajratilgan katologlar ro'yxatini saqlaydi
PS1	Komandalar satriga taklif etilgan tashqi ko'rinish
PS2	Komandalar satriga qo'shimcha taklif etilgan tashqi ko'rinish
SHELL	Joriy foydalanuvchining interpretatorini saqlaydi
TERM	Terminal turini aniqlash
PWD	Joriy katologni saqlaydi
UID	Bajarilayotgan dastur foydalanuvchisining UID (user ID)
SHLVL	Shell-ning keyingi yuklanishida bu o'zgaruvchi 1 birlikka oshadi. Faqat yaratilgan jarayonlar hisobga olinadi.
RANDOM	Bu o'zgaruvchi har bir qayta o'qilganda foydalanuvchi tasodifiy sonlarni oladi

### 1.6-§. Foydalanuvchi bilan muloqot

**Read** operatori. Shell script-da yozilgan dastur uchun foydalanuvchi tomonidan ma'lumotlarni kiritishda maxsus **read** operatori qo'llaniladi.

```
read [o'zgaruvchilar . . . ]
```

**Read** operatori bajarilishi bilan terminal ekranida foydalanuvchi tomonidan ma'lumot kiritish imkoni beriladi. Ma'lumotlar klaviaturadan kiritib bo'lingandan keyin **Enter** tugmasi bosiladi, misol uchun:

```
1 #!/bin/bash
2 #REPLY test
3 #echo "Kompyuter nomini kiriting va
\ "Enter\ " tugmasini bosing :"
4 echo -n 'Kompyuter nomini kiriting va
"Enter" tugmasini bosing :'
```

```
5 read
6 echo "Siz - $REPLY ni tanladingiz"
```

Odatda **read** operatorini qabul qilishdan oldin ekranda savol chiqadi. Bu **echo** dasturi yordamida bajariladi. Agar dasturda **echo** ning **-n** varianti ko'rsatilmasa, u ma'lumotlar chiqarilgandan keyin avtomatik yangi satrga o'tishni qo'shadi. Shuning uchun misoldagi 3-satrdagi izoh berilgan.

4-satrdagi ekranga chiqarish taklifi keltirilgan. Bittalik qo'shtirnoqdagi chiqariladigan satr to'liq olinadi. Bu yerda ikkitalik qo'shtirnoq ichidagi **Enter** so'zi ham e'tiborga olingan.

5-satr. Agar **read** operatori o'zgaruvchini ko'rsatmasdan chaqirilsa, u foydalanuvchi tomonidan kiritilgan barcha ma'lumotlarni jimlik holatida **REPLY** o'zgaruvchisiga ta'minlab beradi.

6-satrdagi ushbu o'zgaruvchida saqlangan ma'lumotlar chiqariladi.

### 1.7-§. O'rniga qo'yish.

Shell script-da ma'lumotlarni o'rniga qo'yishning kuchli vositasi mavjud bo'lib, u dastur chiqaradigan ma'lumotlarni standart asosida chiqaradi.

O'rniga qo'yishni qo'llash uchun dasturda teskari bittalik qo'shtirnoqdan foydalanish kerak. Masalan:

```
`data`
yoki $ belgisi bilan qavsdan foydalaniladi:
$(data)
```

Dastur kodida o'rniga qo'yish qo'llanilgan joyda dastur standart chiqarilishni bajaradi. Bu qiymat dinamik bo'lib, ma'lumotlarni o'rniga qo'yish skript bajarilayotgan vaqtda amalga oshiriladi. Misol uchun quyidagi dastur matnini ko'rib chiqamiz:

```
1 #!/bin/bash
2 #O'rniga qo'yish
3 echo "*****"
4 DATE=`date`
5 echo "DATE=$DATE"
6 echo "*****"
7 USERS=`who | wc=1`
8 echo "tizimdagi USERS=$USERS"
```

```

9 echo "*****"
10 UP=$(data; uptime)
11 echo "Tarkibi=$UP"
12 echo "*****"
13 exit 0

```

4-satrdan **DATE** o'zgaruvchisiga dastur standart chiqarishi kerak bo'lgan **data** qiymati o'zlashtirilgan. 5-satrdan ushbu o'zgaruvchining qiymati chiqariladi.

O'rniga qo'yishda komandalar konveyerini qo'llash mumkin (7-satr). Natijada **wc** dasturiga berilgan standart chiqarish qo'llaniladi.

10-satrdan o'rniga qo'yish bir vaqtda ikkita dastur bilan bajariladi. Natijada **UP** o'zgaruvchisi **data** va **uptime** dasturlarini standart chiqarishini bajaradi.

## 1.8-§. Arifmetik ifodalar

Qobiqqa o'rnatilgan til asosan, fayl tizimlari ob'ektlari bilan amallar bajarishga mo'ljallangan. Shuning uchun ham unda faqat butun sonli arifmetik amallarni bajarish imkoniyati mavjud va arifmetik amallarning minimal to'plami kiritilgan.

Arifmetik ifodalarda  $+$ ,  $-$ ,  $*$ ,  $/$  va qavs operatorlarini qo'llash mumkin. Shuningdek,  $++$  va  $--$  kabi unar operatorlarni ham ishlatiladi.

Arifmetik ifodalarning qiymatlarini o'rniga qo'yish uchun ifodani  $\$$  belgisi bilan boshlanuvchi ikkita qavs ichiga yozish kerak bo'ladi. Masalan,

```
$((2*2))
```

Arifmetik ifodalarda qobiq va tashqi o'zgaruvchilardan foydalanish mumkin. Bunda ushbu o'zgaruvchilarning qiymatlari butun son ekanligi tekshiriladi. Agar o'zgaruvchilar butun qiymatga ega bo'lmasa, arifmetik ifodaning qiymatiga 0 qiymat o'zlashtiriladi. Masalan:

```

1 #!/bin/bash
2 #Arifmetik ifodalarning qiymatini o'rniga
qo'yish
3 PERM=2
4 echo "2*2=$(2*$PERM)"
5 echo "((2*3+5)-4)/2=$(( (2*3+5)4)/2 )"

```

Agar keltirilgan misolda PERM o'zgaruvchisiga (3-satr) belgisi, masalan test qiymati berilsa, dastur natijasi echo (4-satr) ekranga 0 qiymatini chiqaradi.

### 1.9-§. IF shartli operatori

If operatorida shart tekshirilib, dastur kodi son qaytaradi. Agar dastur kodi muvaffaqiyatli bajarilsa, u nol qiymatni qaytaradi. Agar dastur bajarilishida xatolik bo'lsa, noldan farq qiluvchi sonni qaytaradi.

Xuddi shunday shell script-da nol rost qiymat, agar nol bo'lmasa yolg'on qiymat sifatida baholanadi.

if operatori hamma vaqt fi operatori bilan tugallanishi kerak.

Agar tekshiralayotgan shart rost bo'lsa, then va fi so'zlari orasidagi operatorlar ketma-ketligi bajariladi.

Else operatori shart yolg'on bo'lganda bajariladigan operatorlar ketma-ketligini qo'llash uchun foydalaniladi. Quyida shartli operator qo'llanilgan dasturga misol keltirilgan:

```
1 #!/bin/bash
2 # if then else ga misol
3 if rm test>2 /dev/null
4 then
5     echo "Deleted"
6 else
7     echo "Not deleted"
8 fi
```

3-satrda rm dasturining kodini tekshirish berilgan. Bu yerda rm dasturining yuklanishida beriladigan barcha tavsiflar keltirilgan. Agar dastur test faylini o'chira olsa, bu kod qaytaradigan qiymat nolga teng bo'ladi. Agar qaysidir sabablarga ko'ra o'chira olmasa, noldan farqli kodga qaytadi. Buning uchun xatolik bo'yicha xabar chiqarilmaydi, xatolikni standart chiqarish /dev/null ga yo'naltirilgan.

Kodda nolni qaytarish 5-satrda bajariladi. Noldan farqli qiymatni qaytarish – 7-satrda bajariladi.

Agar if operatorida shart sifatida kvadrat qavs ichidagi ifoda qo'llanilsa, bu shartga ruxsat berish uchun test dasturi chaqiriladi. If qaytarish dasturi test ning kodini tekshiradi.

Quyidagi dastur ham oldingi misolda berilgan vazifani bajaradi,

lekin faylning mavjudligi haqidagi shartni tekshirish uchun test dasturini chaqiradi (3-satrdagi shart).

```
1 #!/bin/bash
2 # if then else ga test qo'llashga oid
```

misol

```
3 if [ -w $HOME/bin -a -f $HOME/bin/test ]
4 then
5     rm $HOME/bin/test
6     echo "test deleted"
7 else
8     echo "test not deleted"
9 fi
```



3- satrdagi “[” belgisidan keyin va “]” belgisidan oldin albatta probel qo'yilishiga e'tibor berish kerak. Bu probellar albatta bo'lishi shart.

### 1.10-§. Shartni test dasturi yordamida tekshirish

Dastur ikkita turdagi mantiqiy shartni tekshiradi, bular VA (AND) hamda YOKI (OR).

**Ifoda1 -a Ifoda2** – agar Ifoda1 va Ifoda2 rost bo'lsa, rost qiymat qaytaradi.

**Ifoda1 -o Ifoda2** – agar Ifoda1 yoki Ifoda2 rost bo'lsa rost qiymat qaytaradi.

**!** operatori mantiqiy ifodaning qiymatini qaytaradi.

Sonlarni taqqoslash quyidagi operatorlar yordamida amalga oshiriladi:

**1-son -eq 2-son** – agar berilgan sonlar teng bo'lsa, rost.

**1-son -ne 2-son** – agar sonlar teng bo'lmas, rost.

**1-son -gt 2-son** – agar birinchi son katta bo'lsa, rost.

**1-son -ge 2-son** – agar birinchi son ikkinchi sonda katta yoki teng bo'lsa, rost.

**1-son -lt 2-son** – agar birinchi son ikkinchi sondan kichik bo'lsa, rost.

**1-son -le 2-son** – agar birinchi son ikkinchi sondan kichik yoki teng bo'lsa, rost.

Satrlarni taqqoslash:

**-n satr** – agar satr uzunligi nolga teng bo'lmasa, rost,

**-z satr** – agar star nolga teng uzunlikda bo‘lsa, rost,  
**1-satr=2-satr** – agar 1-satr aynan 2-satrga teng bo‘lsa, rost.  
Satrlarni taqqoslashda belgilarning registri (katta yoki kichik) hisobga olinadi.

Fayl turi va mavjudligini tekshirish:

- ye /fayl/ga/yo`l** – agar fayl mavjud bo‘lsa, rost,
- f /fayl/ga/yo`l** – agar fayl mavjud bo‘lsa va u oddiy (odatdagi) fayl bo‘lsa, rost,
- d /fayl/ga/yo`l** – agar fayl mavjud va u katalog bo‘lsa, rost,
- L /fayl/ga/yo`l** – agar fayl mavjud va u belgili ko‘rsatkich (havola) bo‘lsa.
- r /fayl/ga/yo`l** – agar fayl mavjud bo‘lsa va o‘qish uchun ruxsat berilgan bo‘lsa, rost,
- w /fayl/ga/yo`l** – agar fayl mavjud va yozish uchun ruxsat berilgan bo‘lsa, rost,
- x /fayl/ga/yo`l** – agar fayl mavjud bo‘lsa va bajarilishga ruxsat berilgan bo‘lsa, rost.
- s /fayl/ga/yo`l** – agar fayl mavjud bo‘lsa va nol bo‘lmagan uzunlikka ega bo‘lsa, rost.

### 1.11-§. O‘rnatilgan operatorlar && va || larni qo‘llash

Shell script-da mantiqiy shartlar VA - && hamda YOKI - || larni tekshirish uchun operatorlar mavjud.

Bu operatorlar qaysidir shart rost bo‘lganda bitta komandani bajarilishini yoki yolg‘on bo‘lganda komandaning bajarilmasligini tekshirish uchun qo‘llaniladi. Masalan:

```
[ -f file ] && rm file
```

Ya’ni, ikkala shart ham rost bo‘lganda mantiqiy VA tekshiriladi. Shuning uchun ham agar birinchi shart rost bo‘lsa, u holda ikkinchi shart tekshiriladi, ya’ni rm dasturi chaqiriladi.

```
[ -f file ] || touch file
```

Bu misolda file fayli mavjudligi tekshiriladi. Agar fayl mavjud bo‘lmasa (birinchi shart YOLG‘ON), ushbu faylni yaratish uchun touch dasturi chaqiriladi. ROST qiymatni olish uchun hech



bo‘lmasda shartlarning biri ROST qiymat qaytarishi kerak. Shuning uchun, agar fayl mavjud bo‘lsa (ROST), touch dasturi chaqirilmaydi, ikkinchi shartni tekshirish zarur bo‘lmaydi. Agar fayl mavjud bo‘lmasa (YOLG‘ON), ikkinchi shart tekshiriladi va touch dasturi bajariladi.

### 1.12-§. Case operatori

Case operatori berilgan satrni andozalar bilan navbatma-navbat solishtiradi. Agar andoza mos kelsa, andoza bilan maxsus belgi ;; orasidagi operatorlar ro‘yxati bajariladi. Barcha satrlar bajarilgandan keyin boshqarish esac so‘zidan keyin joylashgan operatorlarga uzatiladi.

Case operatori hamma vaqt esac kalit so‘zi bilan tugallanadi. Ushbu operator qo‘llanilgan dasturga misol:

```
1 #!/bin/bash
2 #case esac ga misol
3 case $TERM in
4     *term)
5         echo "Bu xterm?"
6     ;;
7     unknown |vt [0-9]*)
8         echo "Bu vt100 ?"
9     ;;
10    linux)
11        echo " Bu LINUX terminali!!!"
12    ;;
13    *)
14    echo " Men bu terminalni bilmayman : ("
15    esac
16    exit 0
```

3-satrdan case operatori TERM o‘zgaruvchisidagi satrni tekshirishga beradi.

4-satrdan \*term) andozasi bilan solishtiriladi. Andozalarni yozishda o‘rniga qo‘yish belgilaridan foydalanish mumkin, xuddi shunday fayl nomlarini yozishda ham. Berilgan andozada term bilan tugallanuvchi ixtiyoriy satrdan foydalanish mumkin. Agar andoza ishlasa, u holda 5-satrdagi komanda bajariladi. Agar andoza

ishlamasa, u holda keyingi andozani tekshirishga o'tadi.

7-satrdagi YOKI operatori | belgisi bilan qo'llanilishiga misol keltirilgan. Shunday namuna asosida satr unknown so'ziga mos kelishi yoki u vt va raqamlardan boshlanishi tekshiriladi. Agar andoza ishlansa, 8-satrdagi komandalar bajariladi. Ishlamasa, u holda keyingi andoza tekshiriladi.

10-satrdagi berilgan satr linux so'ziga mos kelishi tekshiriladi. Agar mos kelsa, 11-satrdagi komandalar bajariladi, aks holda keyingi andoza tekshiriladi.

Agar birorta ham andoza mos kelmasa (13-satr), u holda andoza sifatida \* belgisini ishlatish mumkin, bu belgi S dasturlash tilidagi default so'ziga mos keladi.

Quyidagi misolda case operatoridan foydalanishning yana bir imkoniyati keltirilgan.

```
1 #!/bin/bash
2 # case esac misol
3 echo -n "[Y|yes] tugmalarini bosing: "
4 read YN
5 case $YN in
6     [yY]|[yY] [eE] [sS]
7     echo "Siz $YN ni bosdingiz"
8     ;;
9     *)
10    echo "Siz [Y|yes] ni bosmadingiz "
11 esac
12 exit 0
```

Bu misolda foydalanuvchidan Y yoki yes so'zlarini kiritish so'raladi (3-satr). Foydalanuvchining javobi YN o'zgaruvchiga o'zlashtiriladi (4-satr). Case operatori yordamida foydalanuvchi nima kiritganligi tekshiriladi. Bunda harflarning katta kichik yozilishining ahamiyati yo'q, 6-satrdagi andoza turli xil variantlarni tekshirib beradi. Talab qilingan belgilar kiritilsa, 7-satrdagi komanda bajariladi. Agar ixtiyoriy boshqa belgilar kiritilsa, u holda 9 satrdagi andoza jimlik holatida bajariladi va xatolik haqidagi xabar chiqariladi (10-satr).

### 1.13-§. For operatori

For operatorining har bir qadamida ro‘yxatning keyingi qiymati o‘zlashtiriladi va do hamda done orasidagi barcha operatorlar bajariladi.

Operator ro‘yxatdagi barcha qiymatlar qayta ishlanmaguncha yoki tsikl tanasida break operatori uchramaguncha takrorlanadi.

Misol:

```
1 #!/bin/bash
2 # for misol
3 for I in 1 2 3 4 5 6 7 8 9 10
4 do
5 echo "→ $I ←"
6 done
7 exit 0
```

Bu misolda I o‘zgaruvchiga ro‘yxatdagi: 1, 2,3, 4, 5, 6, 7, 8, 9 va 10 qiymatlari qo‘yilgan. Natijada ekranda 10 ta satr chiqariladi.

Ro‘yxat elementlari probel yoki tabulyatsiya bilan ajratiladi. Agar ro‘yxat elementlari bitta satrga sig‘masa, u holda keyingi satrda ham davom ettirish mumkin, keyingi satrga o‘tish uchun Enter tugmasini bosishdan oldin \ belgisi qo‘yib yoziladi. Bu belgi yangi satrga o‘tishni bildiradi. Misol uchun:

```
for I in list1 list2 list3 \
    list4 list5 list6
```

for tsiklida ro‘yxat sifatida fayl tizimining andozasini ko‘rsatish mumkin. Bu holda har bir qadamda o‘zgaruvchiga andozani qanoatlantiruvchi faylga yo‘l ko‘rsatiladi. Ushbu mexanizm qo‘llanilgan misol:

```
1 #!/bin/bash
2 # for misol
3 # html faylni yaratish
4 #for FILES in `ls ~/.bash_*.html`
5 for FILES in ~/.bash_*.html
6 do
7 echo "<HTML>" > $(FILES).html
8 echo "<HEAD><TITLE>$(FILES)</TITLE>
</HEAD>" >>$(FILES).html
9 echo "<BODY><PRE>" >> $(FILES).html
10 cat $(FILES) >> $(FILES).html
```

```
11 echo "</PRE></BODY></HTML>" >>
$(FILES).html
12 chmod a+r $(FILES).html
13 done
14 exit 0
```

5-satrdan ro'yxat sifatida ~/.bash\_\* andozasi qo'llanilgan. Bu FILES o'zgaruvchisiga foydalanuvchining bosh katologida joylashgan nomi .bash\_ bilan boshlanadigan fayl o'zlashtirilganligini bildiradi. Odatda bu shart .bash\_profile, .bash\_logout va .bash\_history fayllariga mos keladi. Ya'ni, tsiklda yuqoridagi ro'yxatdagi fayl nomlari soniga mos ravishda 3 ta qadam bajariladi: FILES o'zgaruvchisi har bir qadamda ro'yxatdagi fayllardan birini o'qiydi.

9-satrdan 11-satrgacha xuddi shunday nomdagi faqat .html kengaytmasi bilan yangi fayl tashkil etiladi. Yaratilgan faylda kiruvchi faylning mazmuni joylashadi.

Ushbu skript bajarilishi natijasida foydalanuvchining bosh katologida .html kengaytmali fayl paydo bo'ladi.

### 1.14-§. Tashqi fayldan ma'lumot olish

Ba'zan shell script dasturiga konfiguratsiya ma'lumotlarini bog'lashga to'g'ri keladi. Buning uchun tashqi fayldan ma'lumot olish zarurati paydo bo'ladi. Tashqi fayldan ma'lumot olishning bir necha usullari mavjud bo'lib, ulardan biri bilan quyidagi misolda qarab chiqamiz. Boshqa usullar bilan ham keyingi misollarda tanishamiz.

Faraz qilaylik, skriptda bir xil mazmuni bo'yicha faqat bitta parametrini o'zgartiradigan bir nechta komandalarni bajarishga to'g'ri kelsin. Masalan, firewall qoidasi bo'yicha bir nechta kompyuterning IP-adresini yozishni olsak, bunda aniq resursga ruxsat olish mumkin bo'ladi. Bu ruxsatni skriptga kiritish uchun dastur satrining ko'rinishi quyidagicha bo'ladi:

```
iptables -A FORWARD -s IP_ADDRESS -j ACCEPT
```

Bizning misolda IP\_ADDRESS o'rniga tashqi faylda joylashgan IP adreslar qiymatlari qo'yiladi. Bunga misol sifatida quyidagi satrlar yozilgan faylni olamiz (fayl nomi – misol-data):

```
1 192.168.0.1
2 192.168.0.2
3 #192.168.0.3
```

```
4 192.168.0.4 # not pay
5 192.168.0.5
```

Bu faylning ba'zi satrlari yoki qism satrlari izoh ko'rinishda berilgan (3- va 4-satrlar). Linux-da izoh berish uchun # belgisi qo'llanilishi qoida sifatida qabul qilingan.

Tashqi fayldan ma'lumotlarni olish bo'yicha misol:

```
1 #! /bin/bash
2 # tashqi fayldan ma'lumot olishga misol
3 for I in `cat ./misol-data`
4 do
5     echo "→ $I"
6 done
7 exit 0
```

Tashqi fayldagi ma'lumotlarga murojaat qilish uchun for tsikli qo'llaniladi. Ma'lumotlar ro'yxati sifatida **cat ./misol-data** o'rniga qo'yish natijasida beriladigan ma'lumotlar olinadi. Natijada ekranga nafaqat IP adreslar, balki ushbu faylda bo'lgan barcha qiymatlar chiqariladi. Gap shundki, shell script tilida satrlar bilan ishlash operatorlari mavjud emas.

To'g'ri natija olish uchun faylda berilgan ma'lumotlarni filtrlash zarur bo'ladi. Buning uchun boshqa dasturdan foydalaniladi. Bunday imkoniyatlardan biri quyidagi misolda berilgan:

```
1 #! /bin/bash
2 # tashqi fayldan ma'lumot olishga misol
3 for I in `cat ./misol-data | cut -f1 -d `
4 | sed -e `/#/ d``
5 do
6     echo "→ $I"
7 done
8 exit 0
```

Bu yerda dastlab, faylning birinchi maydoni tanlanadi, keyin esa # belgisi ishtirok etgan satrlar o'chiriladi.

### 1.15-§. While operatori

While tsiklida do va done kalit so'zlari orasidagi operatorlar

to'plami shart rost bo'lgan holatlarda (yolg'on bo'lmaguncha) yoki break yoki exit operatorlari uchramaguncha bajariladi.

Ushbu tsikl operatori qo'llanilgan oddiy misol:

```
1 #! /bin/bash
2 # while ga misol
3 x=1
4 while [ $x -lt 10 ]
5 do
6     echo "→ $X ←"
7     x=$(( $X+1 ))
8 done
9 exit 0
```

Dastlab X o'zgaruvchiga 1 qiymat beriladi (3-satr). Keyin X ning qiymati 10 dan kichik ekanligi tekshiriladi. Agar shart rost bo'lsa, 6- va 7-satrlar orasidagi amallar bajariladi. 7-satrdan X ning qiymati 1 birlikka oshiriladi va shart yangidan tekshiriladi. Skriptning ishlashi natijasida ekranda 9 ta satrdan iborat ma'lumotlar chiqariladi.

### 1.16-§. SELECT operatori

Select operatori standart chiqarishda nomerlangan ro'yxatni va chiqarilgan elementlarning nomerini kiritish taklifini chiqaradi va foydalanuvchi ma'lumot kiritgandan keyin Enter tugmasini bosishi kerak bo'ladi. Foydalanuvchi tomonidan kiritilgan nomer asosida tanlangan element o'zgaruvchiga ta'minlanadi va do va done orasidagi operatorlar ro'yxati bajariladi. Shundan so'ng yana element nomerini kiritish taklifi beriladi (shell-ning versiyasiga mos ravishda).

Bu tsikldan chiqish uchun break yoki exit operatorlaridan birini oshkora chaqirish kerak.

Select operatorini qo'llash bo'yicha misol:

```
1 #! /bin/bash
2 # select ga misol
3 select FILE in ~/.b* QUIT
4 do
5     if [ -e $FILE ]
6     then
7         ls -l $FILE
8     else
```

```
9          break
10     fi
11 done
12 exit
```

Bu misolda select operatori ro'yxatning nomerlangan qiymatlarini chiqaradi. Ro'yxatga foydalanuvchining ildiz katologida joylashgan .b harfi bilan boshlangan barcha fayllar va QUIT so'zlari kiradi. Ekranda ma'lumot kiritish taklifi chiqarilgandan keyin element nomerini kiritish va Enter tugmasini bosish kerak. Shundan so'ng FILE o'zgaruvchisiga kiritilgan nomerga mos qiymat ta'minlanadi va do va done kalit so'zlari orasidagi amallar bajariladi.

5-satrdagi kiritilgan qiymat bo'yicha "shunday fayl mavjudmi?" sharti tekshiriladi. Bu yerda fayl turi ahamiyatga ega emas. Agar fayl mavjud bo'lsa, ls dasturi (7-satr) bajariladi. Keyin yana ro'yxat yoki ma'lumot kiritish taklifi paydo bo'ladi. Agar fayl mavjud bo'lmasa va QUIT so'ziga mos raqam tanlangan bo'lsa, break operatori bajariladi (9-satr) va select tsikli tanasidan chiqariladi.

### 1.17-§. Nuqta operatori

Nuqta operatori shell script kodida yozilgan boshqa fayldagi kodni bajarish uchun qo'llaniladi.

Tochka operatori o'z xususiyatlari bilan S dasturlash tilidagi include instruktsiyasiga to'g'ri keladi. Lekin boshqa fayldagi shell script faylini qo'shishda uni komandalar satri orqali ham oddiy dasturlarni yuklash kabi berish mumkin.

Nuqta operatori odatda tashqi konfiguratsiya fayllaridagi konfiguratsiya parametrlarini olish uchun qo'llaniladi.

Masalan, quyidagi mazmundagi fayl mavjud bo'lsin:

```
PARAM=value
PARAM2=value2
```

Ushbu parametrlarni boshqa faylda qo'llash uchun, birinchi faylning matnini qo'shish kerak bo'ladi. Qo'shilayotgan fayl bajarilmaydigan bo'lishi mumkin.

```
.file
echo $PARAM
echo $PARAM2
```

Nuqta operatorining qo'llanilishi bo'yicha misol:

```
1 #! /bin/bash
2     #funksiya     va     "."operatorining
qo'llanilishiga misol
3 if [ ! -x $HOME/bin/misol-2 ]; then
4     exit 1
5 fi
6 . $HOME/bin/misol-2
7 select FILE in ~/.* QUIT
8 do
9     if [ -f $FILE ]
10    then
11        any
12    else
13        break
14    fi
15 done
16 exit 0
```

Ushbu misolda keltirilgan skript ham select operatoriga keltirilgan funktsional vazifalarni aynan bajaradi, lekin bunda ikkita qiziqarli xususiyat qo'llanilgan.

3-satrdan 6-satrgacha orqali qo'shiladigan qo'shimcha misol-2 fayli tekshiriladi. Agar u mavjud bo'lmasa, skript ishi dastur bajarilishida paydo bo'ladigan xatolik haqida xabar o'rnida 1-kodli exit operatori orqali yakunlanadi.

misol-2 faylida funktsiyani aniqlash (tavsiflash)ga doir misol ko'rsatilgan:

```
1 #! /bin/bash
2 # funktsiyani aniqlashga misol
3 #function
4 any ()
5 {
6     ls -l $FILE
7 }
```

Shell script-da funktsiya ikki xil usulda aniqlanishi mumkin: birinchisi function operatori yordamida yoki ikkinchisi funktsiya nomidan keyin ochiluvchi va yopiluvchi qavslarni qo'yish orqali.



Funksiya tanasi figurali qavs ichiga olib yoziladi.

Shuningdek, shell script-da barcha o'zgaruvchilar global hisoblanadi, funksiya tanasi ichida ham shell script-ning ixtiyoriy joyida aniqlangan o'zgaruvchilar qo'llanilishi mumkin. Shu bilan birgalikda skriptning ixtiyoriy joyida qo'llash mumkin bo'lgan yangi o'zgaruvchini aniqlash mumkin bo'ladi. Bunda yagona istisno qilish mumkin bo'lgan holat pozitsiyali o'zgaruvchilar hisoblanadi. Bunday o'zgaruvchilar haqida keyingi mavzuda muhokama qilamiz.

Agar funksiyada qandaydir parametr (o'zgaruvchi qiymati) kiritilishi zarur bo'lsa, bu parametr funksiya nomidan keyin komandalar satri argumenti sifatida yoziladi. Funksiya ichida bunday parametrlarni olish pozitsiyali o'zgaruvchilar yordamida amalga oshiriladi.

### 1.18-§. Maxsus o'zgaruvchilar

Shell script-ga o'rnatilgan maxsus o'zgaruvchilar mavjud. Birinchi navbatda – bular pozitsiyali o'zgaruvchilar. Bunday o'zgaruvchilar yordamida komandalar satriga dasturlarni chaqirishdagi parametr qiymatlari olinadi.

**\$0 \$1 . . . \$9** – pozitsiyali o'zgaruvchilar.

**\$0** – dastur nomi

**\$1** – komanda satrining birinchi parametri

**\$2** – komandalar satrining ikkinchi parametri va h.k.

**\$#** - dasturga berilgan komandalar satri parametrlari soni.

Dasturning nomi hisobga olinmaydi.

**\$\* va \$@** - komandalar satrining barcha parametrlari

**\$\* va \$@** o'zgaruvchilar orasida farq qiluvchi jihatlari bor.

Faraz qilaylik, dastur quyidagi namuna bo'yicha yuklangan bo'lsin:

```
program -v -f "The file"
```

Agar ushbu dasturning ichida barcha o'zgaruvchilar ro'yxatini olishga yo'naltirilgan bo'lsa, masalan, for tsiklida quyida keltirilgan satrlardan biri yozilishi mumkin:

```
for I in $* # to`rtta qadam
```

```
for I in $@ # to`rtta qadam
```

```
for I in "$@" # uchta qadam
```

Misoldan ko'rinib turibdiki, qoniqarli natijani albatta ikkitalik qo'shtirnoq ichiga olib yozilgan "\$@" ni qo'llab olish mumkin ekan.

## 1.19-§. Boshqa maxsus o'zgaruvchilar

Pozitsiyali o'zgaruvchilar qo'llanilgan misol:

```
1 #! /bin/bash
2 # $0 ning bajarilishiga misol
3 case $0 in
4   *listtar)
5     echo " $1 arxiv ro\`yxati..."
6     TARGS= "-tvf $1"
7     ;;
8   *maketar)
9     echo " $1.tar arxivini yaratish..."
10    TARGS= "-cvf $1.tar $1"
11    ;;
12  *) echo "Qo\`llanilishi: tar-fayllar
ro\`yxati|maketar dir"
13    exit 88
14 esac
15 tar $TARGS
16 exit 0
```

Linux-da dasturlarga juda ko'p komandalar satri parametrlarini kiritish kerak bo'ladi. Chunki, barcha dasturlar komandalar satrining bir xil parametrlarini uzatadi. Yuklanuvchi dasturda har bir aniq holatga yangi skript yozmaslik uchun dasturni yuklashning quyidagi usulini qo'llash mumkin. Unga bir nechta belgili havolalar kiradi. Dasturni yuklashda ushbu havolalardan foydalanish kerak. Dastur aynan qaysi nom bilan yuklanishi (havola nomi bo'yicha) biladi va ushbu nomga bog'liqlikni zarur funksiyaga avtomatik qo'shiladi.

Shell script-da u aynan qaysi nom bilan yuklanishini belgilash mumkin. Buning uchun \$0 pozitsiyali o'zgaruvchidan foydalaniladi. Keltirilgan misolning 3-satridagi case operatori chaqirilgan dastur nomini nazorat qiladi. Andoza yordamida ikkita mumkin bo'lgan nomlar olinadi: listtar (4-satr) va maketar (8-satr). Ya'ni biz dastur yuklanishida qanday yo'l bilan ko'rsatilishini bilmaymiz, (nom va yo'l \$0 ga tushadimi), andozada esa nom oldiga \* belgisi qo'yilgan.

Misoldagi skript tar arxivini yaratish va uning mazmunini ko'rish uchun mo'ljallangan. Shuning uchun ham listtar holatida dastur ko'riladigan arxiv uchun nom berishi kerak. Bu nom \$1

o'zgaruvchisiga tushadi. 5-satrdan ekranga « \$1 arxivi ro'yxati...» xabari chiqariladi, bunda \$1-ning o'rniga arxiv nomi qo'yiladi. 6-satrdan esa, tar dasturiga beriladigan parametr TARGS o'zgaruvchisiga o'zlashtiriladi. Shundan keyin boshqarish 15-satrga uzatiladi.

Maketar holatida dastur arxiv hosil qiladi. Argument sifatida katalog nomi beriladi. Bu nom \$1 o'zgaruvchisiga tushadi. 9-satrdan ekranga hosil qilinadigan arxivning nomi haqida xabar chiqariladi. 10-satrdan esa, tar dasturiga beriladigan parametrlar shakllantiriladi. Shundan so'ng boshqaruv 15-satrga uzatiladi.

Agar dastur qandaydir boshqa nom ostida chaqirilsa, u holda andoza jimlik holati bo'yicha ishlaydi (12 satr). Buning uchun xatolik haqidagi xabar chiqariladi va 88 kodli xatolik bilan dasturdan chiqib ketadi (13-satr).



*Dastur qanday xatolikni qaytarishini dasturchi hal qiladi.*

Skriptning ishlashi uchun ikkita belgili havola yaratish zarur bo'ladi: listtar va maketar. Uni faqat shu nomlar bilan chaqirish kerak. Agar skript uni nomi bilan chaqirsa - misolda xatolik haqida xabar beriladi.

## 1.20-§. Getopts dasturini qo'llash

Getopts – bu dasturga kiritilgan komandalar satrini ajratib olish uchun qo'llaniladigan komanda.

Bu komanda faqat POSIX uslubida yozilgan parametrlarni tushunadi, ya'ni parametrlar oldiga tire belgisi qo'yilgan bitta harfdan iborat bo'lishi kerak.

Masalan: **-v, -t, -f file** boshq. Belgining parametrini aniqlashda parametrning qo'shimcha qiymatga ega ekanligi bilinadi.

Ushbu stilda yozishga misol:

```
1 #! /bin/bash
2 while getopts f:o:v OPTION
3 do
4 case $OPTION in
5 f) echo "Option f - $OPTARG argumenti" ;;
6 o) echo "Option f - $OPTARG argumenti" ;;
7 v) echo "Option f - argument yo`q" ;;
```

```

8 \?) echo "Usage: `basename $0` -f infile
[-o outfile] [-v]"
9 esac
10 done
11 exit 0

```

Getopts dasturi while tsiklini shartli chaqiradi. U komandalar satrining parametrini qidirishga kirishadi, agar bunday parametr bo'lsa, dastur uni OPTION o'zgaruvchisiga joylashtiradi va rost qiymat qaytaradi.

While tsiklining qadami tugallangandan keyin, getopts dasturi yangidan chaqiriladi. Bu dastur komandalar satrining keyingi parametrini qidiradi, ya'ni keyingi parametr topilsa, birinchi qadamdagi barcha amallar takrorlanadi. Agar parametr topilmasa, u holda dastur yolg'on qiymat qaytaradi va tsikldan chiqadi.

Agar komandalar satrining qo'shimcha parametrlari mavjud bo'lsa, getopts dasturi bu parametrni maxsus OPTARG o'zgaruvchisiga joylashtiradi.

Getopts dasturining bajarilishi vaqtida quyidagi xatoliklar bo'lishi mumkin:

- dasturning parametrlari noaniq ko'rsatilgan
- komandalar satri parametrda majburiy qo'shimcha parametr ko'rsatilmagan.

Agar dasturning ishlashida xatolik yuz bersa, u stderr xatolik haqidagi xabarni chiqaradi, OPTION o'zgaruvchisiga esa ? belgisi o'zlashtiriladi.

## 1.21-§. Trap operatori

Trap operatori dasturning olayotgan signallariga standart reaksiyasini aniqlashda qo'llaniladi.

Birinchi opsiya sifatida signalni olishda bajariladigan komandani ko'rsatish kerak. Komanda sifatida funktsiyani ishlatish mumkin. Keyin signallar ro'yxatini probel bilan ajratib ko'rsatish kerak.

Trap operatorining qo'llanilishi bo'yicha misol:

```

1 #! /bin/bash
2 trap clean 2
3 clean () (

```

```

4      X=1
5      echo "Start formatting /dev/hda3:"
6      while [ $X -lt 10 ]
7      do
8          echo -n ".."
9          sleep 2
10         X=$(( $X+1 ))
11     done
12         echo "Done"
13     exit 0
14 )
15     while [ 0 ]
16     do
17         ;
18     done
19 exit 0

```

2-satrdagi trap operatori dastur 1-signalni qabul qilganda clean funksiyasi bajarilishi uchun chaqiriladi.

Dasturning asosiy tanasida cheksiz takrorlanuvchi while tsikli joylashgan (15-18-satrlarda), chunki, tsikldagi shart har doim rost qiymat qabul qiladi. Dastur hech qachon tugallanmaydi. Ya'ni while tsiklida do va done lar orasida qandaydir operator yozilishi kerak, bu yerda esa ; -bo'sh operator qo'llanilgan.

Clean funksiyasi «Start formatting /dev/hda3:» xabarini ekranga chiqaradi, keyin esa 2 sekundlik ushlanish bilan (9 satrdagi sleep operatori) ikkita nuqtani chiqaradi.

Funksiyaning oxirida exit operatori bajariladi (13-satr) va dastur ishi yakunlanadi.



1. Agar dastur chaqirilganda aniqlanmagan o'zgaruvchiga murojaat bo'lsa, qanday hodisa sodir bo'ladi?
2. if va while operatorlarida shart sifatida nima qaraladi?
3. Qanday operator yordamida skriptning bajarilishi tugallanadi?
4. Qaysi operator boshqa fayldagi kodni qo'shish uchun qo'llaniladi?
5. Maxsus belgilarni ekranga chiqarish uchun qanday belgidan foydalaniladi?
6. Dasturda xatolik yuz berganda qanday xatolik kodi

qaytariladi?

## **2-BOB. LINUXNING INITSIALIZATSIYA (ISHGA TUSHIRISH) TIZIMI**

### **2.1-§. Linux ishga tushirish tizimlari**

Linux ishga tushirish tizimi – bu tizimning yuklanishida bajariladigan skriptlar majmuasi hisoblanadi. Skriptlar klassik Bourne Shell (“sh”) dagi shell-script tilida yoziladi.

Tarixiy nuqtai nazardan olib qaraganda ikkita initsializatsiya tizimi mavjud:

- System V
- BSD

Bu tizimlar bir-biridan yuklovchi skriptlarning tuzilishi bilan farq qiladi: ularning qanday nomlanishi, qaysi katologda joylashishi, chaqiruv ketma-ketligi va boshq.

Linux-da eng ko‘p tarqalgan ishga tushirish tizimi System V hisoblanadi. Bu tizim quyidagi distributivlarda ishlatiladi:

- RedHat Linux
- SuSE Linux
- Mandriva va boshqa ko‘plab distributivlarda.

BSD tizimi esa Slackware Linux va uning avlodlarida ishlatiladi.

Agar System V tizimi haqida gap ketsa, bu juda qat’iy tizim bo‘lib, agar “o‘ngga” yoki “chapga” bir qadam chiqilsa, shu vaqtning o‘zida ish to‘xtatiladi.

BSD tizimida qat’iy qoidalar yo‘q, lekin uni yaratish prinsiplari aniqlangan. Shuning uchun ham agar operatsion tizim va uni ishlatadigan distributivlarni olib qarasak, bu betartib tashkil etilgan deyish mumkin. Ya’ni, fayllarni nomlashga mos keladigan va ularni chaqirish tartibi bir xil bo‘lgan ikkita o‘xshash tizim mavjud emas. Lekin bu qanday ishlashini hamma tushanadi.

Shaxsiy kompyuterlarni ishga tushirganimizda, quyidagi ketma-ketlikdagi amallar bajariladi:

- Kompyuterning BIOS tizimi bajariladi.
- operatsion tizimning yuklovchisi LILO, grub yoki boshq. yuklanadi
- Linux yadrosi yuklanadi.

Yadro ishga tushadi va u asosiy amallardan tashqari ildiz fayl tizimini faqat o‘qish rejimida bog‘laydi, shuningdek, tizimdagi

birinchi dastur init ni ishga tushiradi.

Natijada quyidagilarga ega bo‘lamiz:

- faqat o‘qish rejimida ishlaydigan ildiz fayl tizimiga bog‘lanish.
- init dasturini ishga tushirish.

Tizimning yuklanishida bajariladigan keyingi amallar ko‘p hollarda init dasturining konfiguratsiyasiga bog‘liq bo‘lib qoladi. Shuni e‘tiborga olish kerakki, joriy vaqt momentiga qadar qaysi yuklovchi tizimdan foydalanilgani muhim emas. Bularga bog‘liq bo‘lmagan holda barcha Linux-larda dastlab init yuklanadi. Keyingi qadamlarda qaysi ishga tushirish tizimidan foydalanish init dasturining qanday konfiguratsiya qilinganligiga bog‘liq bo‘lib qoladi.

## 2.2-§. Init dasturi

Yuqorida ta’kidlanganidek, tizimdagi birinchi jarayon bu init dasturi hisoblanadi. Hamda uning qanday konfiguratsiya qilinganligiga qarab keyingi tizimlar yuklanadi.

Linux yadrosi yuklanish vaqtida albatta ildiz fayl tizimini bog‘laydi (odatda faqat o‘qish rejimida). Shuning uchun init dasturi bajarilishi (buning uchun dasturning fayliga kirishga ruxsat olish kerak) va o‘zining konfiguratsiya fayli /etc/inittab-ni o‘qishi mumkin.

Initning yuklanish jarayoni har bir S dasturlash tili bilan tanish bo‘lgan odam init/main.c faylida yadroning dastur kodini ko‘rishi mumkin, qolganlar uchun:

- birinchi jarayon paydo bo‘lish vaqti kelganda, birinchi bajariladigan ish yadro dasturini ko‘rsatilgan rdinit= parametri bilan ishga tushirishga harakat qiladi;

- agar bunday parametr yadroga berilmagan bo‘lsa, u holda /init yuklanadi;

- agar /init bilan hech narsa yuz bermasa, u holda yadro dasturni ko‘rsatilgan init= parametri bilan ishga tushiradi;

- bu parametr bilan ham hech narsa yuz bermasa, u holda yadro /sbin/init ni yuklaydi;

- agar /sbin/init bilan muammo paydo bo‘lsa, u holda yadro /etc/init ni ishga tushiradi;

- /etc/init ishga tushirish ham bajarilmasa, u holda yadro /bin/init ni ishga tushiradi;

- agar /bin/init ni ishga tushirish imkoni bo‘lmasa, yadro /bin/sh ni yuklaydi.

- agar `/bin/sh` ham yuklanmasa, u holda yadro xabarlarini bilan ishlashiga to'g'ri keladi.



*/bin, /sbin, /etc, /dev kataloglarini hech qachon alohida bo'limga o'tkazmang. Ularda joylashgan fayllar tizimning yuklanishida zarur bo'lishi mumkin. Bunga aniq misol sifatida `init` dasturi va uning konfiguratsiya fayli `/etc/inittab` ni keltirish mumkin.*

Linux-da bajarilish darajasi (run level) tushunchasi mavjud. Bajarilish darajasi 0 dan 6 gacha sonlar bilan belgilanadi.

Tizim belgilangan vaqt oralig'ida mos bajarilish darajasida joylashadi. Siz, tizim administratori sifatida uni bitta bajarilish darajasidan boshqasiga o'tkazishingiz mumkin. Bu `init` (yoki `telinit`) dasturi yordamida bajariladi. Buning uchun dasturga argument sifatida bajarilish darajasining mos soni kiritiladi. Masalan, tizimni 3-bajarilish darajasiga o'tkazish uchun `init` dasturini quyidagi ko'rinishda yuklash kerak bo'ladi:

### **init 3**

Linuxning turli distributivlarida bajarilish darajalari turli maqsadlarda qo'llaniladi.

`init` dasturining zamonaviy versiyasi bajarilish darajasining o'nta holatidan foydalanishlari mumkin, lekin odatda yettitasi qo'llaniladi.

- **0** – tizimning o'chirilishi bo'yicha amallar bajarilayapti.
- **1** – bir foydalanuvchi rejimi (single user mode). Tizimni tiklash bo'yicha turli administrator amallari uchun mo'ljallangan. O'z ma'nosi bo'yicha Safe Mode Windows rejimiga o'xshaydi, lekin uni to'liq takrorlanmaydi. Tizimning ushbu bajarilish darajasida to'liq konfiguratsiyalangan, lekin birorta ham xizmat yuklanmagan, foydalanuvchilardan esa, faqat bitta root foydalanuvchisi ishlay oladi.
- **2** – qo'llanilmaydi, lekin 3-bajarilish darajasi kabi konfiguratsiya qilingan. RedHat va SuSE Linux-larda 3-bajarilish darajasi kabi konfiguratsiyalangan, lekin tarmoq fayl tizimining qo'llab-quvvatlovisiz konfiguratsiya qilingan. Debianda ko'p foydalanuvchili rejim sifatida qo'llaniladi.
- **3** – ko'p foydalanuvchili rejim (multiuser mode). Serverning ishlashi uchun me'yordagi rejim.
- **4** - Slackware Linux-da tizimga grafik rejimda kirish uchun qo'llaniladi. RedHat va SuSE Linux-da konfiguratsiya qilinmagan.
- **5** - RedHat va SuSE Linux-da tizimga grafi rejimda kirish



uchun qo'llaniladi. Slackware Linux-da konfiguratsiya qilinmagan.

- 6 – tizimni qayta yuklash bo'yicha amallarni bajarish.

Superfoydalanuvchi tizimni nol bajarilish darajasiga o'tkazish orqali to'xtatishi mumkin:

```
init 0
```

yoki tizimni qayta yuklashi mumkin:

```
init 6
```

### 2.3-§. /etc/inittab faylining formati

Init dasturining konfiguratsiya fayli matnli fayl hisoblanadi. Izoh belgisi - #. Fayl quyidagi satrlardagi formatdan tashkil topgan:

- id - identifikator, bir yoki ikkita belgi. Bu maydon unikal bo'lishi shart. Zamonaviy init dasturi versiyalari ishiga hech qanday ta'siri yo'q.

- run\_level – bajarilayotgan dasturning bajarilish darajalari ro'yxati. Bu maydon bo'sh bo'lishi mumkin.

- action – dasturning bajarilish xususiyatlarini aniqlaydi. Bu maydonga oldindan aniqlangan so'zni yozish mumkin.

- process – bajarilayotgan dastur. Init satrlarni faylda yozilgan tartibda qarab chiqadi. Faraz qilaylik, tizimni uchinchi bajarilish darajasiga o'tkazish kerak va init quyidagi ko'rinishda yuklangan bo'lsin:

```
init 3
```

Dastur faqat run\_level maydonida 3 raqami bo'lgan satrdan bajarilishni boshlaydi. Ya'ni, process maydonida yozilgan dastur ishga tushiriladi, lekin action maydonida aniqlangan ba'zi kelishuvlar bilan ishga tushiriladi.

Madomiki, dasturni insonlar ingliz tili ona tili bo'lgan o'z tillarida yozar ekan, ular dasturga ba'zi ingliz tili grammatikasi xususiyatlarini kiritgan. Ingliz tilida qoidalar mavjud, lekin ko'p sonli bu qoidalarning istisnolari ham mavjud. Oldingi abzatsda bu qoidalar yozilgan, lekin bir qancha istisnolar ham bor.

Keling Slackware Linux distributivi misolida inittab konfiguratsiya faylini ko'rib chiqamiz. Quyida ushbu faylning mazmuni izohlarsiz va bo'sh satrlarsiz keltirilgan.

```
id:3:initdefault:
```

```
si:S:sysinit: /etc/rc.d/rc.S
```

```
su:1$:wait: /etc/rc.d/rc.K
```

```

rc:2345:wait:/etc/rc.d/rc.M
ca: :ctrlaltdel:/sbin/shutdown -t5 -r now
10:0:wait:/etc/rc.d/rc.0
16:6:wait:/etc/rc.d/rc.6
pf::powerfail:/sbin/genpowerfail start
pg::powerokwait:/sbin/genpowerfail stop
c1:1235:respawn:/sbin/agetty 38400 tty1
linux
c2:1235:respawn:/sbin/agetty 38400 tty2
linux
c3:1235:respawn:/sbin/agetty 38400 tty3
linux
c4:1235:respawn:/sbin/agetty 38400 tty4
linux
c5:1235:respawn:/sbin/agetty 38400 tty5
linux
c6:1235:respawn:/sbin/agetty 38400 tty6
linux
x1:4:wait:/etc/rc.d/rc.4

```

Tizimning yuklanishida yadro init dasturini bajarilish darajasini ko'rsatmasdan yuklaydi. Ya'ni, oddiygina

```
init
```

Dastur qandaydir usulda ushbu vaziyatda qaysi satr bajarilayotganligini bilishi kerak. Qanday bajarilish darajasi qo'llanilgan? Bajarilish darajasini ko'rsatish uchun quyidagi satrdan foydalaniladi:

```
id:3:initdefault:
```

Bu umumiy qoidaning birinchi istisnosi. Ko'rganimizdek, process maydoni bo'sh. Action maydonida initdefault kalit so'zi yozilgan bo'lib, bu run\_level maydonida bajarilish darajasi jimlik holati bo'yicha o'rnatilganligini bildiradi.

Endi tizim uchinchi bajarilish darajasida yuklanganligini hisobga olib faylni o'rganib chiqamiz. Quyidagi satrda umumiy qoidalarning birdaniga ikkita istisnosi keltirilgan:

```
si:S:sysinit: /etc/rc.d/rc.S
```

Birinchidan, run\_level maydonida S harfi turibdi va action

maydonida esa sysinit, o‘z-o‘zidan bunday yozilmagan

S harfidan boshlaymiz. Yadroga yuklanishda single parametri beriladi, yadro init dasturini S harfi turgan run\_level maydonidan tashqari barcha satrlarga reaksiyasini to‘xtatgan holda ishga tushiradi. Odatda, distributivlar bu vaziyatda tizim birinchi bajarilish darajasi (single user mode)da yuklanishiga mos konfiguratsiya qilingan bo‘ladi.

Sysinit amali process maydoniga yozilgan dastur faqat tizimni ishga tushirishda bajarilishini anglatadi va init keyingi satrlarni ishga tushirishdan oldin uning to‘xtashini kutadi. Xuddi shunday /etc/rc.d/rc.S dasturi faqat tizim ishga tushayotganda bajariladi. Bir bajarilish darajasidan boshqasiga o‘tish vaqtida u yuklanmaydi.

```
su:ls:wait:/etc/rc.d/rc.K
```

Uchinchi bajarilish darajasidagi satr ishlamaydi, biz uni o‘tkazib yuboramiz.

```
rc:2345:wait:/etc/rc.d/rc.M
```

run\_level maydonida 3 raqami bor, shuning uchun /etc/rc.d/rc.M dasturi yuklanadi. Wait amali init inittab faylidagi keyingi satrlarni bajarishni boshlashdan oldin, dasturning bajarilishi yakunlanishini kutishini anglatadi. Xuddi shunday tizimning ishga tushayotgan vaqtida ikkinchi dastur rc.M yuklanadi.

```
ca: :ctrlaltdel:/sbin/shutdown -t5 -r now
```

Bu satrda run\_level maydonida hech qanday raqam yo‘q, bu tizim ishi boshlanishida ushbu satr bajarilmasligini bildiradi. U holda ishga tushish vaqtida birdaniga shutdown dasturi yuklanishi juda g‘alati. Lekin action maydonida ctrlaltdel parametri mavjud. Bu parametr init-ning Ctrl(chap)+Alt(chap)+Del tugmalar kombinatsiyasi bosilishidan keyin bajarilishini bildiradi. Agar kimdir ushbu tugmalar kombinatsiyasini bossa, process maydonida ko‘rsatilgan dastur bajariladi.



*Foydalanuvchi tizimdan ro‘yxatdan o‘tgan yoki o‘tmaganligi muhim emasligiga e‘tibor bering. Ixtiyoriy inson ushbu tugmalar kombinatsiyasini bosib yuborishi mumkin, natijada tizim qayta yuklanadi.*

```
10:0:wait:/etc/rc.d/rc.0
```

```
10:6:wait:/etc/rc.d/rc.6
```

Bu ikki satr bajarilishining uchinchi darajasiga umuman ta'sir qilmaydi, unga bog'liq emas, shuning uchun biz uni tashab o'tamiz.

```
pf::powerfail:/sbin/genpowerfail start  
pg::powerokwait:/sbin/genpowerfail stop
```

Bu satrlarda ham run\_level maydoni bo'sh, tizim ishga tushishida ular bajarilmaydi. Lekin init ularni eslab qoladi. Bu satrlar bajarilishi uchun sizning kompyuteringizda intellektual UPS ulangan bo'lishi kerak. Shuningdek, UPS komandalarini tushinadigan dastur ishga tushadi (odatda genpowerd dasturida qo'llaniladi).

Agar elektr toki uzilib qolsa, u holda UPS dasturga bu haqida xabar beradi, o'z navbatida init powerfail amalini qayta ishlash uchun ma'lumot beradi. Agar elektor tokidagi uzilish tiklansa, powerokwait amalini qayta ishlaydi.

Genpowerfail dasturi – bu oddiy skript. Agar u start parametri bilan ishga tushsa, u holda shutdown dasturi ishga tushiriladi. UPSning holatiga bog'liq holda tizim ishini yakunlash protsedurasi darhol bajariladi yoki bir yoki ikki minut kechikish bilan amalga oshiriladi. Agar skript stop parametri bilan ishga tushsa, u holda tizim ishini yakunash protsedurasi inkor qilinadi

**(shutdown -s) .**

Jadvalda action maydonida qo'llash mumkin bo'lgan ba'zi kalit so'zlari keltirilgan.

<b>Maydon</b>	<b>Tavsifi</b>
initdefault	Tizimning yuklanish vaqtida jimlik holati bo'yicha bajarilish darajasini aniqlaydi
sysinit	Tizimning yuklanishida dasturning eng birinchi bajarilishini aniqlaydi. init dasturlar ro'yxatidagi keyingi dasturni ishga tushirishdan oldin uning yakunlanishini kutadi.
wait	Dastur bir marta yuklanadi. init ro'yxatdagi keyingi dasturni yuklashdan oldin uning yakunlanishini kutadi.
once	Dastur bir marta yuklanadi. init uning yakunlanishini kutadi.
ctrlaltdel	Ctrl+Alt+Del tugmalar kombinatsiyasi bosilgandan keyin ishga tushadigan dasturni aniqlaydi.

powerfail	init jarayoni elektr tokining uzilishi signalini olganda ishga tushadigan dasturni aniqlaydi.
powerokwait	init jarayoni elektr tokining uzilish qayta tiklangan signalini olganda ishga tushadigan dasturni aniqlaydi
respawn	Jarayon ishga tushadi, init jarayonning yakunlanishini kutmaydi va satrlar ro'yxatidagi keyingisini qayta ishlashni boshlaydi. Agar jarayon o'z ishini yakunlasa, init uni yangidan yuklaydi.

Konfiguratsiya faylidan xulosa qilish mumkinki, 3-bajarilish darajasida init quyidagi dasturlarni yuklaydi:

- /etc/rc.d/rc.S
- /etc/rc.d/rc.M
- har bir virtual konsol uchun /sbin/agetty.

## 2.4-§. BSD initsializatsiya tizimi

Slackware Linux-da oddiy ishga tushiruvchi skriptlardan farq qiluvchi, BSD initsializatsiya tizimiga o'xshash initsializatsiya tizimi qo'llaniladi. Shu oddiylikdan kelib chiqib, misol sifatida Slackware **Linux-ning** ishga tushiruvchi skriptini ko'rib chiqamiz.

Barcha skriptlar /etc/rc.d katologida joylashgan.

**/etc/rc.d/rc.S fayli.** /etc/rc.d/rc.S fayli quyidagilarga mo'ljallangan:

- swap fazosiga ulanish.
- fayl tizimini tekshirish.
- fayl tizimiga ulanish.
- modullarni yuklash.

/etc/inittab fayliga asosan, eng birinchi bajarilish darajasiga bog'liq bo'lmagan holda /etc/rc.d/rc.S dasturi ishga tushadi. Dastur shell script tilida yozilgan va uning kodi bo'yicha tizimning ishga tushishida qanday amallar bajarilishini bilib olish juda oson.

Quyidagi misolda Slackware Linux distributividan rc.S skripti keltirilgan, tizimning yuklanish bosqichlarini to'liq ko'rib chiqamiz.

**#!/bin/sh**

Birinchi satrda dasturning bajarilishi uchun qo'llaniladigan interpretator aniqlangan.

```
PATH=/sbin:/usr/sbin:/bin:/usr/bin
```

PATH o'zgaruvchisining aniqlanishi. Bu o'zgaruvchi bajariladigan dasturni ishlash kerak bo'lgan katalogga yo'lni saqlaydi.



*Bu o'zgaruvchi ushbu skript uchun lokal bo'lib, uning qiymati tizimdagi qolgan dasturlar uchun o'rinli bo'lmaydi. PATH o'zgaruvchisining dastlabki qiymati maxsus konfiguratsiya faylini aniqlaydi.*

```
/sbin/mount -v proc /proc -n -t proc
```

Keyin proc fayl tizimi /proc katalogiga ulanishi amalga oshiriladi. Shundan keyin fayl tizimida taqdim etilgan barcha imkoniyatlarga ruxsat beriladi.

«-n» parametri /etc/mtab fayliga satr qo'shishni inkor qilishga e'tibor bering.

```
if [ -w /proc/sys/kernel/hotplug ]; then
  if grep -w nohotplug /proc/cmdline 1>
/dev/null 2> /dev/null ; then
  echo "/dev/null" > /proc/sys/kernel
/hotplug
  elif { " -x /etc/rc.d/rc.hotplug ]; then
  echo "/dev/null" > /proc/sys/kernel
/hotplug
  fi
fi
```

Bu satrlarda hotplug tizimini yuklash imkoniyatlari tekshirish amalga oshiriladi.

Hotplug tizimining asosiy vazifasi – kompyuterga elektr tokini ajratmasdan ulangan qurilmalar uchun drayverlarini yuklash. Bunday qurilmalarga PCMCIA, PCI va USB shinalariga ulanadigan qurilmalar kiradi.

Dastlab /proc/sys/kernel/hotplug faylining mavjudligini va bu faylga yozish imkoniyatlari tekshiriladi. Faylda Linux yadrosi tomonidan bajariladigan yangi qurilmaning ulanishini aniqlagan dasturlarni yuklash satri yozilgan. Agar bu fayl bo'lmasa, yadro qurilmalarning ulanishini hisobga olmasdan ishga tushirilganligini bildiradi va tizim ishga tushishida drayverlarni avtomatik yuklashning

hojati bo'lmaydi.

Agar Linux yadrosi qurilmalarning ulanganligini hisobga olish imkoniyatini qo'llab-quvvatlasa, lekin sizda qandaydir sabablarga ko'ra uni ishlatish xohishi bo'lmasa, u holda Slackware Linux-da hotplug quyidagi ikkita usulda o'chirilishi mumkin:

1. Yadroning ishga tushishida unga komandalar satrida hotplug paramerini beradi. Birinchi if operatori ushbu parametrning /proc/cmdline faylida mavjudligini tekshiradi. Agar parametr mavjud bo'lsa, /proc/sys/kernel/hotplug faylining mazmuni o'chiriladi va yadro yangi qurilma paydo bo'lganda qaysi dasturni ishga tushirishni bilmaydi.

2. Bajariluvchi bo'lmagan /etc/rc.d/rc.hotplug fayli hosil qilish. Elif operatori ushbu faylini tekshiradi. Agar uni bajariluvchi bo'lmagan fayl bo'lsa, /proc/sys/kernel/hotplug faylining mazmuni o'chiriladi.

```
if [ -x /etc/rc.d/rc.devfsd ]; then
    /etc/rs.d/rc.devfsd start
fi
```

Keyin /etc/rc.d/rc.devfsd faylining mavjudligi tekshiriladi. Bu fayl devfsd virtual fayl tizimini qo'llab-quvvatlashni qo'shuvchi skript hisoblanadi. Zamonaviy Linux tizimlarida bu sistemani keyinchilik qo'llashga tavsiya etilmadi.

```
if [ -d /sys ]; then
    if cat /proc/filesystems | grep -n sysfs 1>
/dev/null 2> /dev/null
    then
        if ! cat /proc/mounts | grep -w sysfs 1>
/dev/null 2> /dev/null
            then
                /sbin/mount -v sysfs /sys -n -t sysfs
            fi
        fi
    fi
```

Keyingi satrlarda sysfs virtual fayl tizimiga ulanish imkoniyati tekshiriladi. Uni qo'llab-quvvatlash **Linux-ning** 2.6.x versiyasidan boshlangan yadrolarda paydo bo'lgan. Ushbu fayl tizimida tizimga ulangan barcha qurilmalarning parametrlari tasvirlanadi.

Sysfs fayl tizimiga ulanish mumkin:

- agar /sys katologi mavjud bo'lsa.
- agar sysfs ni yadro qo'llab-quvvatlasa (qo'llab-quvvatlovchi fayl tizimlarining ro'yxati /proc/filesystems faylida joylashgan).

- ulangan fayl tizimi ro'yxatda bo'lmasa (ulangan fayl tizimlari ro'yxati /proc/mounts faylida joylashgan).

```
if [ -x /etc/rc.d/rc.udev ]; then
    if ! grep -w nchotplug /proc/cmdline 1>
/dev/null; then
        /etc/rc.d/rc.udev
    fi
fi
```

Udev dasturining yuklanish imkoniyati tekshiriladi. Bu dastur qurilma fayllarini dinamik shakllantirish uchun mo'ljallangan. Qurilmalar fayllari tizimda mavjud bo'lgan qurilmalar uchun hosil qilinadi. Qurilmalar haqidagi ma'lumotlar manbasi sifatida sysfs fayl tizimi qo'llaniladi.

Udev dasturini yuklash uchun maxsus ishga tushirish skripti /etc/rc.d/rc.udev qo'llaniladi. Shuning uchun ham birinchi navbatda ushbu faylning mavjudligi tekshiriladi. Skript albatta bajariluvchi bo'lishi shart.

Agar Linux hotplug tizimini qo'llamasdan ishga tushirilsa, u holda udev ni ishga tushirish kerak bo'lmaydi, chunki, udevning ishlashi to'g'ridan-to'g'ri hotplug bilan bog'liq.

```
/sbin/swapon -a
```

/etc/fstab faylida tavsiflangan barcha swap turidagi fayl tizimlari va swap fayllari bog'lanadi (ulanadi).

```
READWRITE=no
if touch /fsrwtastfile 2>/dev/null; then
    rm -f /fsrwtastfile
    READWRITE=yes
else
    echo "Testing root filesystem status: read-
only filesystem"
fi
```

Yadro yuklangandan keyin Linux ildiz fayl tizimiga bog'lanadi



(ulanadi). Ushbu satrlarda fayl tizimlariga qanday rejimda bog‘lanish tekshiriladi: faqat o‘qish uchun yoki to‘liq kirish rejimida.

/fsrwtestfile fayli hosil qilinganligi tekshiriladi. Agar hosil qilingan bo‘lsa, tizim rw rejimida ulangan va READWRITE o‘zgaruvchisiga yes qiymati ta‘minlanadi. Agar hosil qilinmagan bo‘lsa, ekranga xabar chiqariladi va o‘zgaruvchi eski (oldingi) no qiymati bilan qoladi.

```
if [ -r /etc/forcefsck ]; then
    FORCEFSCK="-f"
fi
```

/etc/forcefsck faylining mavjudligi tekshiriladi. Agar fayl mavjud bo‘lsa, FORCEFSCK o‘zgaruvchisiga “-f” qiymati ta‘minlanadi.

Agar tizim yuklanishida albatta fayl tizimini tekshirishi shart bo‘lsa, /etc/forcefsck faylini hosil qilish yetarli bo‘ladi. Keyingi navbatda, rc.S faylida fsck dasturi FORCEFSCK o‘zgaruvchisida joylashgan parametrini qabul qiladigan fayl tizimini tekshirish uchun chaqiriladi.

Quyida ildiz fayl tizimini tekshirish uchun qo‘llaniladigan kodning katta qismi keltirilgan.

```
if [ ! $READWRITE = yes ]; then
    RETVAL=0
```

Agar ildiz fayl tizimi faqat o‘qish uchun rejimida ulangan bo‘lsa (READWRITE o‘zgaruvchisining qiymati tekshiriladi), u holda if operatorining birinchi qismi bajariladi.

```
if [ ! -r /etc/fastboot ]; then
    echo "Checking root filesystem:"
    /sbin/fsck $FORCEFSCK -C -a /
    RETVAL=$?
fi
```

/etc/fastboot faylining bor yoki yo‘qligi tekshiriladi. Agar fayl mavjud bo‘lmasa, ildiz fayl tizimini tekshirish bajariladi.

Fastboot faylini administrator **Linux-ning** fayl tizimini tekshirmasdan tez yuklanishi uchun ishlatadi. Ya‘ni, bu faylni hosil qilish, fayl tizimini tekshirmasdan tizimning yuklanishi uchun yetarli. fastboot fayli forcefsck faylining oldida ustunlikka ega.

```

if [ $RETVAL -ge 2 ]; then
    if [ $RETVAL -ge 4 ]; then
        echo
        PS1="(Repair filesystem) \#"; export PS1
        sulogin
    else
        echo
    fi
    echo "Unmounting file system."
    /sbin/umount -a -r
    /sbin/mount -n -o remount,ro /
    echo "Rebooting system."
    sleep 2
    reboot
fi

```

Keyin **fsck** dasturining qaytish kodi tekshiriladi. Agar **fsck** dasturi fayl tizimini tekshirish vaqtida hech qanday xatolik topmasa, u 0 qiymat qaytaradi. Agar xatoliklar mavjud bo'lsa va dastur uni tuzatsa, u 1 qiymatni qaytaradi. Agar xatoliklar tuzatilgandan keyin kompyuterni qayta yuklash talab qilinsa, 2 qiymat qaytariladi. Agar dastur xatoliklarni avtomatik tuzata olmasa va administrator aralashishi talab qilinsa, u 4 yoki undan katta qiymatni qaytaradi.

Birinchi if operatori qiymat qaytarish kodi 4 ga teng yoki undan katta ekanligi shartini tekshiradi. Agar shart rost (to'g'ri) bo'lsa ekranga katta xabar chiqariladi (misoldagi kodda bu ko'rsatilmagan). PS1 o'zgaruvchisiga mos qiymat ta'minlanadi va u eksport qilinadi. Keyin root foydalanuvchisining parolini so'raydigan sulogin dasturi yuklanadi va shell dasturi ishga tushiriladi. Ushbu shell da administrator ildiz fayl tizimini tiklash bo'yicha barcha amallarni bajaradi va shell dasturidan chiqadi.

Agar qaytarish kodining qiymati katta yoki teng 2, lekin 4 dan kichik bo'lsa, demak, faqat tizimni qayta yuklash talab etiladi. else dan keyingi operatorlar ishlaydi va ekranga tizimni qayta yuklash haqidagi xabar chiqariladi (keltirilgan misolda xabar ko'rsatilmagan).

echo "Unmounting file system." bilan boshlangan barcha satrlar qaytarish kodi katta yoki teng 2 bo'lganda hamma vaqt bajariladi. Ildiz fayl tizimidan boshqa barcha fayl tizimlari o'chiriladi (yopiladi). Ildiz fayl tizimi faqat o'qish uchun rejimiga o'tadi. 2 sekundlik

kechikish (ushlanib qolish) dan keyin tizim qayta yuklanishi boshlanadi.

```
/sbin/mount -w -v -n -o remount /
  if [ $? -gt 0 ] ; then
    echo
    read junk;
  fi
```

Agar ildiz fayl tizimini tekshirishdan keyin kompyuterni qayta yuklash talab etilmasa, u holda fayl tizimini to'liq kirish rejimiga o'tkazuvchi mount dasturi ishga tushiriladi.

Keyin mount dasturining qaytarish kodi tekshiriladi. Agar xatolik yuz bersa, ekranga xabar chiqariladi (misolda xabar matni keltirilmagan) va tizim foydalanuvchining Enter tugmasini bosishini kutib turadi.

Agar ildiz fayl tizimi boshidanoq to'liq kirish rejimida ulangan bo'lsa, u holda else dan keyingi operatorlar bajariladi.

```
else
  echo "Testing root filesystem status:
read-write filesystem"
  if cat /etc/fstab | grep '\ / ' | grep
umsdos \
1> /dev/null 2> /dev/null ; then
ROOTTYPE="umsdos"
fi
if [ ! "$ROOTTYPE" = "umsdos" ]; then
echo
echo -n "Davom etish uchun ENTER-ni bosing
"
read junk;
fi
fi
```

Birinchi navbatda ildiz fayl tizimining turi tekshiriladi. Agar umsdos fayl tizimi qo'llanilgan bo'lsa, u holda ROOTTYPE o'zgaruvchisiga umsdos qiymati ta'minlanadi. Agar ildiz fayl tizimining turi umsdos turida bo'lmasa, u holda ekranga xatolik haqidagi xabar chiqariladi va foydalanuvchining Enter tugmasini bosishini kutib turadi.



*umsdos fayl tizimi Linux-ni FAT fayl tizimi ustiga o'rnatish uchun qabul qilingan. Hozirgi vaqtda qattiq disklarning o'lchamini hisobga olgan holda bu fayl tizimi amaliyotda qo'llanilmaydi.*

```
/bin/rm -f /etc/mtab*  
/sbin/mount -w -o remount /
```

Fayl tizimiga ulanishda mount dasturi mtab fayliga satr qo'shadi. Fayl tizimini o'chirish(yopish)da umount dasturi mos satrni o'chiradi. Agar tizim noto'g'ri o'chirilsa, ushbu faylda yolg'on ma'lumotlar qolib ketishi mumkin. Shuning uchun ham tizim ishini boshlashda mtab fayli o'chiriladi.

Lekin mtab fayli bir nechta dasturlar ishlashi uchun zarur bo'ladi. Shuning uchun ham mount dasturi ushbu faylni yana bir marta yuklaydi va ulangan fayl tizimiga ushbu fayl tizimining tavsifi mtab fayliga qo'shilishi uchun qayta bog'laydi. Ya'ni, bu fayl yangidan hosil qilinadi.

```
if [ -d /proc/sys ]; then  
    /sbin/mount -f proc /proc -t proc  
fi  
if [ -d /sys/bus ]; then  
    /sbin/mount -f sysfs /sys -t sysfs  
fi
```

Virtual proc va sysfs fayl tizimlarining tavsiflari ham /etc/mtab faylida keltirilgan bo'lishi kerak. Bu fayl tizimlari oldindan ulanishi mumkinligiga qaramasdan, ular qayta ulanishi faqat mos yozuvlar mtab faylida paydo bo'lishi uchun amalga oshiriladi.

```
if ( -x /sbin/hwclock ] ; then  
if grep "^UTC" /etc/hardwareclock 1> /dev/null  
2> /dev/null ; then  
    echo "Setting system time from the hardware  
clock (UTC) ."  
    /sbin/hwclock --utc -hctosys  
else  
echo "Setting system time from the hardware  
clock (localtime) . "  
/sbin/hwclock -- localtime --hctosys  
fi
```

```
fi
```

Agar hwclock dasturi o'rnatilgan bo'lsa, tizimli plataga o'rnatilgan soatdan tizimli vaqtni o'qish amalga oshadi va tizimda vaqt o'rnatiladi.

```
if [ -r /etc/isapnp.conf ]; then
    if [ -x /sbin/isapnp ]; then
        /sbin/isapnp /etc/asapnp.conf
    fi
fi
```

Agar isapnp.conf konfiguratsiya fayli va isapnp dasturi mavjud bo'lsa, oxirgisi ishga tushadi.

```
if [ -x /etc/rc.d/rc.modules -a -r
/proc/modules ]; then
    . /etc/rc.d/rc.modules
fi
```

/proc/modules faylining mavjudligi tekshiriladi. Agar ushbu fayli mavjud bo'lsa, Linux yadrosi yuklanayotgan modullarni qo'llab-quvvatlashga sozlanadi. Bu yerda bajariluvchi rc.modules faylining mavjudligi ham tekshiriladi. Agar ikkala shart ham to'g'ri bo'lsa, u holda rc.modules fayli asosiy fayl rc.S mavjud bo'lgan bash nusxasida bajariladi.

Slackware Linux-da rc.modules tizimning ishga tushishida modullarni yuklash uchun mo'ljallangan. Faylning tarkibini keyinroq ko'rib chiqamiz.

```
if [ -x /sbin/sysctl -a -r /etc/sysctl.conf ];
then
    /sbin/sysctl -e -p /etc/sysctl.conf
fi
```

sysctl dasturi va uning konfiguratsiya fayli sysctl.conf-ning mavjudligi tekshiriladi. Agar ular mavjud bo'lsa, sysctl dasturi yuklanadi.

Sysctl dasturi /proc/sys katologidagi fayllarga parametrlarni yozish uchun qo'llaniladi. Bu dasturlarning o'rniga fayllarga qiymatlarni to'g'ridan-to'g'ri yozish mumkin. Masalan, quyidagicha:

```
echo 1 > /proc/sys/net/ipv4/ip_forward.
```

```
LVM ni yuklovchi satrlar tushirib qoldirilgan.  
if [ ` -r /etc/fastboot ]; then  
    /sbin/fsck $FORCEFSCK -C -R -A -a  
fi
```

Agar /etc/fastboot fayli mavjud bo'lmasa, /etc/fstab faylida tavsiflangan qolgan fayl tizimlarini tekshirish faollashtiriladi.

```
/sbin/mount -a -v -t nonfs,nosmbfs,noproc
```

Tarmoq fayl tizimlari (smbfs va nfs) va proc fayl tizimlaridan tashqari /etc/fstab faylida tavsiflangan barcha fayl tizimlari ulanadi.

```
/sbin/swapon -a
```

Har ehtimolga qarshi yana bir marta swap ulanadi.

```
[ cd /var/log/setup/tmp && rm -rf * ]  
/bin/rm -f /var/run/utmp /var/run/*pid  
/etc/nologin /var/run/lpd* \  
/var/run/ppp* /etc/forcefsck /etc/fastboot
```

Tizimni o'rnatishdan keyin qolgan vaqtincha fayllarni o'chirish jarayoni amalga oshiriladi. Shuningdek, turli xil vaqtinchalik fayllar o'chiriladi. Jumladan, /etc/forcefsck va /etc/fastboot fayllari.

```
if [ -d /initrd ]; then  
    /sbin/umount /initrd 2> /dev/null  
    rmdir /initrd 2> /dev/null  
    blockdev -flushbufs /dev/ram0 2>  
/dev/null  
fi
```

Agar yadro yuklanishida inird (Initial RAM disk) texnologiyasini qo'llagan bo'lsa, RAM diski yopiladi, vaqtinchalik katalog initrd o'chiriladi va RAM disk uchun ajratilgan xotira tozalanadi.

```
touch /var/run/utmp  
chown root.utmp /var/run/utmp  
chmod 664 /var/run/utmp
```

Oldin o'chirilgan utmp fayli hosil qilinadi. Bu faylda joriy vaqtda tizimda bo'lgan foydalanuvchilar ro'yxati saqlanadi. Keyin u

root foydalanuvchisi va utmp guruhiga beriladi va unga kirish huquqi o'zgartiriladi.

Umsdos fayl tizimi deyarli amaliyotda ishlatilmaganligi uchun keyingi satr tashab o'tilgan.

```
echo "$ (/bin/uname -sr) . " > /etc/motd
```

/etc/motd faylida tizim haqidagi ma'lumotlar va Linux yadrosining nomeri joylashadi. Faylning tarkibi foydalanuvchi tizimga kirgandan keyin chiqariladi va uning qo'llanilishi haqida keyinchalik so'z yuritamiz.



*/etc/motd fayli tizimning har bir yuklanishidan keyin hosil qilinishiga e'tibor bering.*

```
if [ -x /etc/rc.d/rc.sysvinit ]; then
    . /etc/rc.d/rc.sysvinit
fi
```

Slacware Linux ishga tushirish skripti SystemV ni qo'llash imkonini beradi. Uni ishga tushirish uchun /etc/rc.d/rc.sysvinit skripti qo'llaniladi. U SystemV ishga tushirish tizimini emulyatsiya qiladi.

```
if [ -f /etc/random-seed ]; then
    echo "Using /etc/random-seed to initialize
/dev/urandom."
    cat /etc/random-seed > /dev/urandom
fi
if [-r /proc/sys/kernel/random/poolsize ]; then
    dd    if=/dev/urandom    of=/etc/random-seed
count=1 \
    bs=$(cat /proc/sys/kernel/random/poolsize)
2> /dev/null
else
    dd    if=/dev/urandom    of=/etc/random-seed
count=1 bs=512 2> /dev/null
fi
chmod 600 /etc/random-seed
```

Faylning oxirgi satrlari tasodifiy sonlar generatori uchun zarur bo'lgan barcha narsalarni hosil qiladi.

**/etc/rc.d/rc.modules va /etc/rc.d/rc.netdevise fayllari.**  
/etc/rc.d/rc.modules fayli tizimning ishga tushishida modullarni

yuklash uchun xizmat qiladi. modullarni yuklash “qo‘lda”, ya’ni oshkor ravishda modprobe dasturini chaqirish orqali amalga oshiriladi.

Quyida rc.modules faylidan satrlar misol sifatida keltirilgan.

```
#!/sbin/modprobe 3c503
```

Bu faylning deyarli barcha satrlari izoh bilan keltirilgan. Tizim yuklanishida sizni qiziqtirgan modul yuklanishi uchun unga mos keluvchi satrni toping va undagi izoh belgisini o‘chiring, agar zarurat bo‘lsa, yuklash parametrlariga qo‘shimcha kiring.

rc.modules faylida rc.netdevice fayliga ulanish amalga oshiriladi:

```
if [ -x /etc/rc.d/rc.netdevice ]; then  
  . /etc/rc.d/rc.netdevice  
fi
```

rc.netdevices fayli tizimning ishga tushishida tarmoq kartasi modulini yuklash uchun qo‘llaniladi. Odatda u o‘rnatish dasturi yordamida hosil qilinadi.

/etc/rc.d/rc.S fayli tizimda bajarilish darajasiga bog‘liq bo‘lmagan holda eng birinchi yuklanadi.

Bu fayl quyidagilar uchun mo‘ljallangan:

- swap fazosiga ulanish.
- fayl tizimlarini tekshirish.
- fayl tizimlariga ulanish.
- rc.modules faylida tavsiflangan qurilmalar drayverini yuklash.
- tizim ishga tushishida zarur bo‘ladigan boshqa ba’zi amallar.

Shunday qilib, rc.S skripti bajarilgandan keyin, xizmatlarni yuklash uchun barchasi tayyor ekanligini aytish mumkin.

/etc/rc.d/rc.M fayli tizimning ikkinchi, uchinchi va to‘rtinchi bajarilish darajasiga o‘tishida bajariladi.

Quyida rc.M faylining tarkibi to‘liq ko‘rib chiqiladi.

```
echo "Going multiuser..."
```

Ekranga ko‘p foydalanuvchili bajarilish darajasiga o‘tganligi haqidagi xabar chiqariladi.

```
/bin/setterm -blank 15 -powersave powerdown -  
powerdown 60
```

Terminallarni o‘rnatuvchi setterm dasturi chaqiriladi.

```
if [ -r /etc/HOSTNAME ]; then  
  /bin/hostname $(cat /etc/HOSTNAME | cut -f1
```



```

-d .)
else
    echo "first.exemple.net" > /etc/HOSTNAME
    /bin/hostname darkstar
fi

```

/etc/HOSTNAME faylining mavjudligi tekshiriladi. Agar fayl mavjud bo'lsa, kompyuter nomini o'rnatadigan hostname dasturi ishga tushiriladi. Kompyuter nomi HOSTNAME faylidan olinadi. Agar fayl mavjud bo'lmasa, u hosil qilinadi. Unga kompyuter nomi first.exemple.net yozib qo'yiladi. Kompyuterning o'ziga first nomi beriladi.

/etc/HOSTNAME fayli tizimni ishga tushirishda muhim ahamiyatga ega. Bu faylda kompyuterlarning to'liq domen nomlari (FQDN) saqlanadi. Agar fayl kompyuter yuklanayotganda tasodifan o'chib ketsa, kompyuterga noto'g'ri nom ta'minlanadi. Shuning uchun rc.M faylidagi first.exemple.net nomini kompyuterning haqiqiy nomi bilan almashtirish tavsiya etiladi.

```

/bin/dmesg -s 65536 > /var/log/dmesg

```

Tizimning ishga tushishida yadro chiqaradigan xabarlar /var/log/dmesg fayliga joylashtiriladi.



*Dastur faqat yadro xabarlarini ko'rsatishiga e'tibor berish kerak. Ishga tushiruvchi skriptlarning xabarlari ko'rsatilmaydi.*

```

if [-x /etc/rc.d/rc.syslog -a -x /usr/sbin
/syslogd -a -d /var/log ]
then
    . /etc/rc.d/rc.syslog start
fi

```

Agar /etc/rc.d/rc.syslog va /usr/sbin/syslogd bajariluvchi fayllari mavjud bo'lsa va shuningdek, /var/log katologi mavjud bo'lsa, /etc/rc.d/rc.syslog ishga tushiruvchi skripti bajariladi. Bu skript tizimga SYSLOG faylini yuklaydi.

```

if [ -x /etc/rc.d/rc.pcmcia ] ; then
    . /etc/rc.d/rc.pcmcia start
    if [ -r /var/run/cardmgr.pid ] ; then
        sleep 5
    fi
fi

```

```
fi
fi
```

Agar /etc/rc.d/rc.pcmcia bajariluvchi fayli mavjud bo'lsa, u holda u bajariladi. Bu ishga tushiruvchi skriptning PCMCIA kontrollerini qo'llab quvvatlashini yoqish uchun mo'ljallangan.

```
if [ -x /etc/rc.d/rc.inet1 ]; then
    . /etc/rc.d/rc.inet1
fi
```

/etc/rc.d/rc.inet1 bajariluvchi faylning mavjudligi tekshiriladi. Agar u mavjud bo'lsa, skript bajariladi. rc.inet1 tarmoq qurilmalari konfiguratsiyalari va marshrutlash jadvali uchun mo'ljallangan. Uning qo'llanilishini keyingi bobda o'rganib chiqamiz.

```
if [ -x /etc/rc.d/rc.hotplug -a -r
/proc/modules ]; then
    if ! grep nohotplug /proc/cmdline 1>
/dev/null 2> /dev/null ; then
        echo "Activating hardware detection:
/etc/rc.d/rc.hotplug start"
        . /etc/rc.d/rc.hotplug start
    fi
fi
```

Bajariluvchi /etc/rc.d/rc.hotplug va /proc/modules fayllarining mavjudligi tekshiriladi. Shuningdek, cmdline faylining (bu faylda yadro ishga tushganda berilgan komandalar satri joylashgan) tarkibida hotplug parametrining yo'qligi tekshiriladi. Agar barcha shartlar to'g'ri bo'lsa, u holda tizimga o'rnatilgan qurilmalarni tekshirish va ularga mos drayverlarni yuklashga mo'ljallangan rc.hotplug skripti yuklanadi.

```
if [ -x /etc/rc.d/rc.inet2 ]; then
    . /etc/rc.d/rc.inet2
fi
```

Agar bajariluvchi /etc/rc.d/rc.inet2 fayli mavjud bo'lsa, u bajariladi.

Ishga tushiruvchi rc.inet2 skripti turli xil tarmoq xizmatlarini yuklash va tarmoq fayl tizimiga ulanish uchun qo'llaniladi.

```
/bin/rm -f /var/lock/* var/spool/uucp/LCK.. * \  
/tmp/.X*lock /tmp/core 2> /dev/null
```

Turli xil vaqtinchalik fayllar o‘chiriladi.

```
chmod 755 / 2> /dev/null  
chmod 1777 /tmp /var/tmp
```

Ildiz katalogga kirish huquqi o‘zgartiriladi, shuningdek /tmp va /var/tmp kataloglariga sticky o‘rnatiladi.

```
if [ -x /sbin/ldconfig ]; then  
echo "Updating shared library linke:  
/sbin/ldconfig"  
/sbin/ldconfig  
fi
```

Bajariluvchi ldconfig fayli mavjudligi tekshiriladi. Agar u mavjud bo‘lsa, dastur bajariladi.

Ldconfig dinamik bibliotekalarning keshini hosil qilish uchun mo‘ljallangan. Bu dastur haqida keyingi bo‘limlarda to‘liq to‘xtalib o‘tamiz.

```
if [ -x /usr/X11R6/bin/fc-cache ]; then  
echo "Updating X font indexes:  
/usr/X11R6/bin/fc-cache"  
/usr/X11R6/bin/fc-cache"  
fi
```

Agar bajariluvchi fc-cache mavjud bo‘lsa, u bajariladi. Fc-cache X Window tizimida shriftlar ideksini yangilash uchun qo‘llaniladi.

```
if [ -x /etc/rc.d/rc.dnsmasq ]; then  
/etc/rc.d/rc.dnsmasq start  
fi
```

Agar bajariluvchi rc.dnsmasq fayli mavjud bo‘lsa, u bajariladi va unga start parametri beriladi. Bu skript keshlovchi DNS serverini yuklash uchun mo‘ljallangan.

```
if [ -x /etc/rc.d/rc.cups ]; then  
/etc/rc.d/rc.cups start  
elif [ -x /etc/rc.d/rc.lprng ]; then  
. /etc/rc.d/rc.lprng start
```

**fi**

Qanday chop qilish (pechat) tizimi cups yoki LPRng oʻrnatilganligini tekshiradi. Tekshirish natijasiga bogʻliq holda, mos skript ishga tushadi.

```
#if [-x /usr/sbin/sartd ]; then
# /usr/sbin/smartd
#fi
```

Agar S.M.A.R.T. texnologiyasidan foydalanishni qoʻshish zarur boʻlsa, yuqorida berilgan satrlardagi izoh belgisini oʻchiring. Bunda tizimning ishga tushishida mos dastur yuklanadi.

```
#if [ -x /sbin/genpoward ]; then
# echo "starting genpoward daemon..."
# /sbin/genpoward
#fi
```

Genpoward dasturi tizimni "intellektual" UPS bilan ishlashiga qoʻl keladi. Agar undan foydalanish zarurati boʻlsa, keltirilgan satrlardagi izoh belgisini oʻchirib chiqing.

```
if [ -x /sbin/accton -a -r /var/log/pacct ];
then
    /sbin/accton /var/log/pacct
    chmod 640 /var/log/pacct
    echo "Process accounting turned on."
fi
```

Accton dasturi ishga tushirish imkoniyatlari tekshiriladi. Bu «BSD process accounting» deb ataluvchi tizimda jarayonlardan nazorat qilish uchun moʻljallangan. Uni qoʻshish (ishga tushirish) uchun Unix yadrosida qoʻllab-quvvatlandigan /var/log/pacct log fayli mavjud boʻlishi kerak.



*Diqqat! «BSD process accounting» tizimining log faylida juda koʻp sondagi maʼlumotlar joylashgan. Shunga eʼtibor berish kerakki, u fayl tizimini toʻldirib yubormasligi kerak.*

```
if [ -x /usr/sbin/crond ]; then
/usr/sbin/crond -110>>/usr/sbin/crond 2>41
fi
```

Cron dasturining ishga tushish imkoniyatlarini tekshirish amalga oshiriladi. Agar dastur o'rnatilgan bo'lsa, u holda yuklanadi. CRON tizimi haqida keyingi bo'limda to'liq ko'rib chiqamiz.

```
if [ -x /usr/sbin/atd ]; then
/usr/sbin/atd -b 15 -l 1
fi
```

Agar bajariluvchi atd fayli mavjud bo'lsa, u tizimning ishga tushishida yuklanadi. Atd dasturi dasturlarning oldindan belgilangan vaqtda bajarilishini ta'minlab beradi.

```
if grep -q quota /etc/fstab ; then
    if [ -x /sbin/quotacheck ]; then
        echo "Checking filesystem quotas:
/sbin/quotacheck -avugn"
        /sbin/quotacheck -avugn
    fi
if [ -x /sbin/quotaon ]; then
    echo "Activating filesystem quotas :
/sbin/quotaon -avug"
    /sbin/quotaon -avug
fi
fi
```

Yuqorida keltirilgan satrlar disk maydonini kvotalash mexanizmini qo'shish (ishga tushirish) imkonini beradi.

Linux-da kvota faqat disk bo'limlariga qo'llanilishi mumkin. Kvotani qo'llashda disk bo'limlari montirovka parametrlari usrquota va grpquota bilan qo'shilgan bo'lishi kerak. Birinchi if operatori bo'limlarning ko'rsatilgan montirovka parametrlari bilan mavjudligini tekshiradi.

Kvotani qo'shishdan oldin fayl tizimining joriy holatini tekshiruvchi quotacheck dasturini ishga tushirish kerak. Bu dastur foydalanuvchining fayllari qancha joy egallashini va bu ma'lumotlarni maxsus fayllarda saqlashni tekshiradi.

Keyin kvotani qo'llab-quvvatlash mexanizmi ishga tushiriladi. Buning uchun mos parametrlar bilan quotaon dasturi qo'llaniladi.

Disk sohasini kvotalashni sozlash haqida keyinroq so'z yuritamiz.

```
if [ -x /etc/rc.d/rc.sendmail ]; then
. /etc/rc.d/rc.sendmail start
fi
```

Agar bajariluvchi /etc/rc.d/rc.sendmail fayli mavjud bo'lsa, u yuklanadi.

rc.M faylidagi qolgan barcha dasturlarning yuklanishi sendmail ning yuklanish prinsiplari bo'yicha bajariladi. Dastlab, mos ishga tushiruvchi skriptlarning mavjudligi tekshiriladi va u mavjud bo'lsa yuklanadi.

Tizimning ishga tushishida eng oxirida rc.local fayli yuklanadi. U Slackware Linux initsializatsiya tizimida standart ishga tushirish skripti bo'lmaganda ilovalarni yuklash uchun mo'ljallangan.

**/etc/rc.d/rc.inet1 fayli** tarmoq interfeysilari va matrshrutlar jadvalining konfiguratsiyasi uchun mo'ljallangan.

Tarmoq interfeyslari konfiguratsiyasi uchun qo'shimcha etc/rc.d/rc.inet1.conf fayli qo'llaniladi. Bu faylning tarkibi yetarlicha sodda, unda to'rtta tarmoq interfeysini konfiguratsiyalash uchun to'rtta guruhga tegishli o'zgaruvchilar mavjud:

```
IPADDR[0]="192.168.0.1"
NETMASK[0]="255.255.255.0"
USE_DHCP[0]=""
DHCP_HOSTNAME[0]=""
```

Shuningdek, jimlik holati bo'yicha marshrutni aniqlash uchun o'zgaruvchi qo'llaniladi:

```
GATEWAY="10.1.1.1"
```

Ushbu faylda yana simsiz tarmoq interfeysini konfiguratsiyalash imkoniyatini beruvchi o'zgaruvchi joylashgan. Lekin, jimlik holati bo'yicha ular izoh belgisi bilan berilgan.

rc.inet1 skriptining kamchiligi uning cheklanganligi hisoblanadi. Masalan, rc.inet1 yordamida tamroq intefeysida qo'shimcha IP adresni aniqlash taqiqlanadi. Tarmoq interfeysida to'rttadan ko'p konfiguratsiyalash mumkin emas. Marshrutlash jadvaliga qo'shimcha marshrutni qo'shish qo'lda bajariladi. Buning uchun rc.inet1 faylining tarkibini to'liq o'chirish mumkin va uni yangidan yoziladi. Skriptni yozishda quyidagilarni hisobga olish kerak:

- Barcha tarmoq interfeyslarining konfiguratsiyasi ifconfig

dasturi yordamida amalga oshadi. Buning uchun har bir interfeysning konfiguratsiyasi uchun ifconfig dasturini barcha zarur parametrlari bilan chaqirish imkonini beruvchi bitta satr yozish kerak.

- Marshrutlash jadvaliga yangi yo‘nalish (marshrut)ni qo‘shish uchun route dasturi qo‘llaniladi. Buning uchun har bir qo‘shiladigan marshrut uchun route dasturi barcha zarur parametrlari bilan chaqirish imkonini beruvchi bittadan satr yozish kerak bo‘ladi.

- Tarmoq interfeyslarini konfiguratsiyalash (sozlash)da unga ulangan tarmoq marshruti marshrutlar jadvaliga avtomatik qo‘shiladi. Route dasturini oshkor chaqirish talab etilmaydi.

- lo tarmoq interfeysini konfiguratsiyalashda marshrutlash jadvaliga 127.0.0.0 tarmog‘i avtomatik qo‘shilmaydi. Shuning uchun ham bu tarmoqni oshkor ravishda qo‘shish kerak:

```
route add -net 127.0.0.0/8 lo
```

Faraz qilaylik, Sizda bitta 192.168.0.1 IP adresi bilan eth0 tarmoq interfeysi mavjud bo‘lsin. Jimlik holati bo‘yicha marshrut – 192.168.0.100. U holda rc.inet1 fayli quyidagi ko‘rinishni oladi:

```
#!/bin/bash  
/sbin/ifconfig lo 127.0.0.1  
/sbin/ifconfig eth0 192.168.0.1  
/sbin/route add -net 127.0.0.0/8 lo  
/sbin/route add default gw 192.168.0.100
```

Agar qandaydir tarmoqqa yangi marshrutlarni qo‘shish kerak bo‘lsa, bu marshrutlar rc.inet1 fayliga route dasturini chaqirish yo‘li bilan tarmoq interfeysini sozlashdan keyin darhol faylning ixtiyoriy joyidan qo‘shish mumkin.

**/etc/rc.d/rc.inet2 fayli** Linuxning ishga tushishida tarmoq ilovalarini yuklash va tarmoq fayl tizimlariga bog‘lanish uchun qo‘llaniladi. Bu faylda tarmoq fayl tizimlari nfs va smbfc larga bog‘lanishga urinish amalga oshiriladi.

Keyin quyidagi ishga tushirish skriptlari (agar zarur bo‘lsa) yuklanadi:

- /etc/rc.d/rc.firewall – firewall sozlamasi.



*Aslida ushbu skript tarmoq interfeyslarining konfiguratsiyasidan oldin rc.inet1 faylining boshida ishga tushirilishi kerak. rc.inet2 faylida esa skript chaqiruvini o‘chirish kerak bo‘ladi.*

- /etc/rc.d/rc.ip\_forward – bu skript paketlarni bitta tarmoq interfeysidan boshqasiga uzatish uchun ruxsat beradi.
- /etc/rc.d/rc.inetd – tarmoq supervizorini yuklash.
- /etc/rc.d/rc.sshd - sshd dasturi yuklash.
- /etc/rc.d/rc.bind - DNS serverni yuklash.
- va boshqa dasturlar.



*Tarmoq xizmatlariga tegishli bo'lgan ba'zi dasturlar rc.M faylida ishga tushiriladi. Misol uchun sendmail va samba dasturilari.*

Slackware Linux-ning ishga tushirish tizimi /etc/rc.d katologida joylashgan ishga tushirish skriptini qo'llash orqali qurilgan.

Agar qandaydir xizmatni yuklamoqchi bo'lsak, unga mos skript bajariluvchi faylga aylantirish kerak bo'ladi. Va aksincha, agar tizim ishga tushishida xizmatni yuklash kerak bo'lmasa, u bajariluvchi skript bo'lishi shart emas.

Barcha ishga tushiruvchi skriptlar o'zini shell script-da yozilgan oddiy dasturni taqdim etadi. Ularning ba'zilari komandalar satridagi parametrlarni tushinadi: - start – yuklash va stop – xizmatni to'xtatish.

Agar Slackware Linux-ning ishga tushirish tizimida qandaydir dasturlarni yuklash uchun ishga tushirish skriptlari hisobga olinmagan bo'lsa, u holda quyidagi ikkita usuldan foydalanish mumkin:

1. /etc/rc.d/rc.local faylida tavsiflangan dasturlarni ishga tushirish.

2. O'zingizning shaxsiy ishga tushirish skriptlaringizni yozing va ularning yuklashni /etc/rc.d/rc.M faylida tavsiflang.

Ikkinchi vaziyatda ishga tushirish skripti quyidagicha ko'rinishda bo'lishi mumkin:

```
#!/bin/bash
start()
(
    echo "Program started"
    program_start
)
stop()
(
    echo "Program stoped"
    killall program
)
```



```

)
case $1 in
    start)
        start ;;
    stop)
        stop ;;
    restart)
        stop
        sleep 2
        start ;;
*)
    echo          "Usage:          Program
start|stop|restart"
    esac

```

/etc/rc.d/rc.M faylida skriptni chaqirish quyidagi ko‘rinishda tashkil etiladi:

```

if [ -x /etc/rc.d/rc.script ]; then
    . /etc/rc.d/rc.script start
fi

```



*Diqqat! Nuqta operatori yordamida faqat shell script tilida yozilgan dasturni bajarish mumkin. Boshqa dasturlarni nuqta operatorisiz chaqirish kerak.*

Lekin eng muhimi – siz ixtiyoriy ishga tushirish skriptiga o‘zgartirish kiritishingiz mumkin. Tizimni yangilashda (Update) SystemV ishga tushirish tizimini qo‘llovchi distributivlardan farq qiluvchi distributivlarda ishga tushirish skriptlari o‘zgarmaydi. Joriy ishga tushirish skriptlarining yonida yangi skriptlar .new kengaytmasi bilan hosil bo‘ladi.

## 2.5-§. SystemV initsializatsiya tizimi

SystemV ishga tushirish tizimi, BSD tizimidan farqli o‘laroq boshqacha prinsipda qurilgan. Linux distributivlari orasida SystemV tizimi quyidagi distributivlarda ham qo‘llaniladi: RedHat Linux, SuSE Linux, Debian, ASPLinux. Bu distributivlarda ham eng birinchi bo‘lib init dasturi yuklanadi. Lekin /etc/inittab konfiguratsiya faylida boshqa faylni chaqiradi.

SystemV ishga tushirish tizimini ASPLinux distributivi misolida

ko‘rib chiqamiz.

/etc/rc.d/rc.sysinit fayli bajarilish darajasiga bog‘liq bo‘lmagan holda eng birinchi bajariladi. Bu dastur shell script tilida yozilgan bo‘lib, administrator mustaqil ravishda unga o‘zgartirishlar kiritishi mumkin.

Faylda quyidagi amallar bajariladi:

- agar initrd texnologiyasi qo‘llanilsa, initrd foydalangan katalog yopiladi va RAM sifatida ajratilgan disk xotirasi tozalanadi.

- yadro xususiyatlarini konfiguratsiyalovchi sysctl dasturi bajariladi.

- tizimli vaqt o‘rnatiladi.

- kompyuter nomi o‘rnatiladi.

- USB va IEEE1394 qurilmalarining ishlashi uchun barcha modullar yuklanadi.

- fayl tizim tekshiriladi.

- fayl tizimiga ulanish bajariladi.

- fayl tizimiga cheklovlar (kvota) qo‘yiladi.

- swap maydonini qo‘llab-quvvatlash qo‘shiladi.

- Agar tizimda dasturiy RAID qo‘llanilsa, uni qo‘llab-quvvatlash qo‘iladi.

- Tizimning qayta yuklanishidan keyin qoladigan ba‘zi fayllar o‘chiriladi.

- IDE interfeysini konfiguratsiyalash uchun hdparm dasturi yuklanadi.

- shuningdek, yana bir qancha amallar.

Keltirib o‘tilgan ro‘yxatdan tizimning dastlabki yuklanishi bajariladi.

Aslida ASP Linux distributivi to‘g‘ridan-to‘g‘ri RedHat Linuxning avlodi hisoblanadi. Shuning uchun ham bu distributivda RedHat ishga tushirish skriptlarining xususiyatlari qo‘llaniladi – tizimning yuklanish konfiguratsiyasi /etc/sysconfig katalogida joylashgan tashqi konfiguratsiya fayllari yordamida amalga oshiriladi.

Konfiguratsiya fayllari “.” Operatori yordamida asosiy skript tanasiga qo‘shiladi. Bu faylda qiymatlari asosiy skriptda tekshiriladigan va qo‘llaniladigan turli xil o‘zgaruvchilar aniqlanadi.

Misol uchun, kompyuter nomini /etc/sysconfig/network faylidan olinadi. Bu faylda HOSTNAME o‘zgaruvchisini aniqlash va unga kompyuterning FQDN nomini ta’minlash mumkin. Network faylining o‘zi esa asosiy skriptga quyidagi ko‘rinishda qo‘shiladi:

```

if [ -f /etc/sysconfig/network ]; then
    . /etc/sysconfig/network
else
    NETWORKING=no
fi

```

Misoldan ko‘rinib turibdiki, dastlab faylning mavjudligi tekshirib ko‘riladi va faqat u mavjud bo‘lgandagina qo‘shiladi. Skriptga qo‘shilgandan keyin network faylida joylashgan o‘zgaruvchilardan foydalanish mumkin. Xost nomining o‘zi esa action funksiyasini chaqirish orqali amalga oshiriladi:

```

action "$Setting hostname ${HOSTNAME} : "
hostname ${HOSTNAME)

```

Ishga tushirish skriptida uchratadigan barcha funksiyalar /etc/rc.d/init.d/functions (belgili havolani qo‘llamasdan faylga haqiqiy yo‘l) faylida joylashgan. Bu fayl ham network faylida ulangani kabi qo‘shiladi.

```

. /etc/rc.d/init.d/functions

```



*Ishga tushirish skriptida /etc/rc.d/init.d/functions faylining mavjudligi tekshirilmaydi, lekin bu fayl ALBATTA bo‘lishi shart.*

/etc/rc.d/rc.sysinit faqat tizimning dastlabki yuklanishida bajariladi. Keyinchalik esa bajarilish darajasiga qarab bajarilish darajasining parametri sifatida beriladigan rc dasturi yuklanadi.

rc dasturining vazifasi- ko‘rsatilgan bajarilish darajalarida ishlashi shart bo‘lgan dasturlarni ishga tushiradi va to‘xtatadi (agar u ishlayotgan bo‘lsa).

BSD ishga tushirish tizimida rc.sysinit skriptining analogi bo‘lishi mumkin, lekin u yerda rc ning analogi yo‘q. Aynan rc ga o‘xshash fayllarning ishga tushirish skriptlari katalogida mavjudligi va xizmatlarni yuklash uchun ishga tushirish skriptlarining yozilishi xususiyatlari SystemV tizimining farqli ekanligini bildiradi.

Agar barcha ishga tushirish skriptlari joylashgan /etc/rc.d<sup>2</sup> katalogining tarkibini qarsak, quyidagi fayl va kataloglarni ko‘rishimiz mumkin:

<sup>2</sup> SUSE Linux distributivida ishga tushirish skriptlari /etc/init.d katalogida joylashgan. RedHat Linux-da ham xuddi shunday katalog mavjud, lekin u /etc/rc.d joylashgan katalogga belgili havola hisoblanadi.

```

    init.d/   rc0.d/   rc2.d/   rc4.d/   rc6.d/
rc.sysinit
    rc       rc1.d/  rc3.d/  rc5.d/  rc.local

```

Bu katologda rc0.d dan boshlanib, rc6.d bilan tugallangan bir qancha o‘xshash katologlar mavjud.

Quyida rc3.d katologining tarkibi keltirilgan.



*Ushbu katologlarda qanday fayllar joylashishini tizim administratori aniqlaydi.*

```

K05saslauthd   K15httpd   K20nfs   K34dhcrelay
K35dhcpcd K35smb K35vncserver K35winbind K36lisa
K45named K50snmpd K50snmptrapd K50vsftpd K54pxe
K65kadmin   K65kprop   K65krb524   K65krb5kdc
K70aep1000 K70bcm5820 K74ntpd K84bgpd K84ospf6d
K84ospfd   K84ripd   K84ripngd   K85zebra
K95firstboot S05kudzu S08iptables S10network
S12syslog S13portmap S14nfslock S17keytable
S20random S24pcmcia S25netfs S26apmd S28autofs
S55sshd S56rawdevices S56xinetd S80postfix
S85gpm S90crond S90cups S90xfs S91dictd
S95anacron S95atd S97rhnsd S99local

```

Bu katologlarda joylashgan fayllar ba’zi umumiy xossalarga ega:

- ularning nomlari “K” yoki “S” bilan boshlanadi.
- birinchi harfdan keyin ikkita raqam bo‘ladi.
- barcha fayllar - /etc/rc.d/init.d katologidagi fayllarga belgili havolalar bo‘lishi yoki shell script formatidagi fayl bo‘lishi mumkin.

Rc dasturining vazifasiga quyidagilar kiradi:

- u rc<bajarilish darajasi nomeri>.d katologiga o‘tadi.
- dastlab K farqi bilan boshlanuvchi barcha fayllar yuklanadi (*ularga stop parametri beriladi*).
- keyin S harfi bilan boshlanuvchi barcha fayllar yuklanadi (*ularga start parametri beriladi*).

/etc/rc.d/init.d katologida joylashgan barcha fayllar – bular ma’lum dasturlarning ishga tushiruvchi skriptlari hisoblanadi.

System V initsializatsiya tizimining ishga tushiruvchi skriptlari albatta start va stop parametrlarini qo‘llab quvvatlashi shart. Bu parametrlarining nomlanishi bo‘yicha start – bu dasturni yuklash, stop – bu dasturni yakunlash (to‘xtatish) deb tushinish mumkin.



*Ishga tushirish skriptlari quyidagi qo‘shimcha parametrlarni ham qo‘llab-quvvatlashlari mumkin: restart, status va boshq. Lekin bu parametrlar majburiy emas.*

Bulardan ko‘rinadiki, rc dastlab dasturlarning bajarilishini ishga tushiruvchi skriptlarning stop parametri bilan bajarishi orqali to‘xtatadi. Keyin esa ishga tushiruvchi skriptlarning start parametri bilan bajarilishiga mos dasturlarni yuklaydi.

Fayl nomidagi raqam skriptning bajarilish tartibini aniqlaydi.

Tizim administratori tizimning ishga tushish vaqtida yuklanadigan dasturlarni o‘zgartirishi, ularni bajarilish darajasiga mos keluvchi katologlarga qo‘shishi va ularni o‘chirib tashlashi mumkin.

## 2.6-§. Chkconfig dasturi

ASPLinux-da RedHat, SuSE va Mandriva Linux-lardagi kabi ishga tushiruvchi skriptlar bilan ishlashni osonlashtirish uchun chkconfig dasturi biriktirilgan.



*Chkconfig dasturi RedHat Linux, Fedora Core, SuSE Linux distributivlarida ham qo'yilgan. SuSE Linux-da dasturning parametrlari farq qiladi.*

Chkconfig dasturi quyidagi amallarni bajarishga imkon beradi:

- qaysi bajarilish darajasida qanday servis (xizmat)lar yuklanishini ko'rsatish.

- chkconfig dasturi boshqarish mumkin bo'lgan servislarni qo'shish va o'chirish imkonini beradi.

- aniq bajarilish darajasida ishga tushirish skriptlarning bajarilishini qo'shish yoki o'chirish imkonini beradi.

Ishga tushirish skriptlari chkconfig dasturi ular bilan ishlashi uchun maxsus tayyorlangan bo'lishi kerak. Buning uchun ularga ikkita satr yozish kerak, ular izoh belgisi bilan bo'lishi shart<sup>3</sup>.

Birinchi satr chkconfig dasturi qaysi bajarilish darajasida skriptni ishga tushirishi, shuningdek, minimal va maksimal ustunlik sonini aniqlaydi.

```
# chkconfig: 2345 20 80
```

Ikkinchi satr – bu dasturning qisqa tavsifi.

```
# description: Saves and restore configuration files  
jf service
```

Qaysi bajarilish darajasida qanday skript bajarilishi kerakligini ko'rish uchun `--list` parametridan ishga tushirish skripti nomini ko'rsatib yoki ko'rsatmasdan foydalanish kerak bo'ladi<sup>4</sup>:

```
chkconfig --list  
chkconfig -list httpd
```

Aniq bajarilish darajasiga ishga tushirish skriptining bajarilishini qo'shish uchun `--level` parametridan foydalanish kerak.

```
chkconfig - --level 35 httpd on
```

<sup>3</sup> SUSE Linux-da chkconfig dasturi parametrlarini ko'rsatish uchun boshqa parametrlar qabul qilingan.

<sup>4</sup> SUSE Linux-da chkconfig dasturiga boshqa parametrlar qo'llaniladi. O'zingiz ishlayotgan distributivdagi dastur hujjatiga qarang.

Agar “on” kalit so‘zi o‘rniga “off” qo‘llanilsa, u holda ko‘rsatilgan skript berilgan bajarilish darajasida ishlamaydi.

Ko‘pchilik hollarda xizmatlarning boshlang‘ich skriptlarini o‘zingiz yozishingiz shart emas, chunki ularning aksariyati tayyor skriptlar bilan beriladi. Sizning asosiy vazifangiz ushbu skriptlarning zarur bajarilish darajasida ishlashiga imkon berishdan iborat.

ASPLinux-da serviceconf dasturi qo‘yilgan. Bu ishga tushirish skriptlarini boshqarish imkonini beruvchi grafik qobiq. Shuningdek, ishga tushirish skriptlarini WEB interfeyslar va webmin servisi yordamida boshqarish mumkin.

## 2.7-§. Xizmatlarni qo‘lda yuklash va to‘xtatish

ASPLinux-da administrator xizmatlarni qo‘lda ishga tushirish va to‘xtatishi mumkin. Buning uchun quyidagilardan foydalaniladi:

- SystemV initsializatsiya tizimning standart ishga tushirish skripti.

- service dasturi.

Birinchi holatda komandalar satriga ishga tushirish skriptiga to‘liq yo‘lni ko‘rsatgan holda yuklash komandasini yozish kerak bo‘ladi. Parametr sifatida ishga tushirish skripti qo‘llab quvvatlaydigan start, stop yoki boshqa parametrlarni kiritish mumkin.

Masalan, ASPLinux-da tarmoqni qo‘llab quvvatlashni yuklash uchun quyidagi komandani kiritish kerak:

```
/etc/rc.d/init.d/network start
```

To‘xtatish uchun:

```
/etc/rc.d/init.d/network stop
```

Qayta ishga tushirish uchun:

```
/etc/rc.d/init.d/network restart
```

Agar service dasturini qo‘llasak, u holda unga parametrlar sifatida xizmatlar va amallarni ishga tushirish skriptining nomi beriladi.

Yuqorida sanab o‘tilgan amallarni service dasturi yordamida bajarish uchun quyidagi komandalardan foydalaniladi:

```
service network start
```

```
service network stop
```

## **service network restart**

Eng asosiysi service dasturi ko'rsatilgan skriptni chaqiradi va unga ko'rsatilgan parametрни beradi.

SuSE Linux-da service dasturi yo'q. Bu yerda boshqa prinsip qo'llaniladi. /sbin yoki /usr/sbin katologida xizmatlarni ishga tushirish skriptiga havola hosil qilinadi. Havolalar rc harflari bilan boshlanib, keyin mos keluvchi ishga tushirish skriptining nomi yoziladi. Misol uchun, rcnetwork.

```
rcnetwork start
```

```
rcnetwork stop
```

## **2.8-§. /etc/sysconfig katologidagi konfiguratsiya fayllari**

/etc/sysconfig katologida ishga tushirish skriptlarining konfiguratsiya fayllari joylashgan. Ushbu fayllarda turli xil o'zgaruvchilar aniqlangan. Fayllarning o'zi esa skriptlarga "." operatori yordamida biriktiriladi.

*Dasturlarning konfiguratsiya fayllari bilan almashtirib yubormang. Misol uchun, Apache WEB serveri va ishga tushirish skriptining konfiguratsiya fayli /etc/rc.d/init.d/httpd. Birinchisi, /etc/httpd/httpd.conf - WEB serverning o'zini konfiguratsiya qilish uchun mo'ljallangan. Ikkinchisi, /etc/sysconfig/httpd httpd dasturi yuklanganda beriladigan parametrlarni ko'rsatish uchun mo'ljallangan.*



/etc/sysconfig katologida fayllar bo'yicha hujjatlar mavjud. Ularni /usr/share/doc/initscripts-<versiya> katologidan topish mumkin. sysconfig.txt faylida barcha mumkin bo'lgan o'zgaruvchilar va ularning qiymatlari tavsifi keltirilgan. SuSE Linux-da konfiguratsiya fayllari bo'yicha hujjatlar mavjud emas, lekin parametrlarning o'zi ushbu fayllarda izohlar orqali yomon yozilgan.

Syscvinifiles faylida ishga tushirish skriptlarini qurish prinsiplari haqida yozilgan. Shuningdek, ishga tushirish skriptlarida foydalanish mumkin bo'lgan funksiyalar haqida ham yozilgan.

## **2.9-§. Root foydalanuvchisining parolini tiklash**

Linux distributivlarining eski versiyalarida super



foydalanuvchining parolini yo‘qotib qo‘yish muammosi osongina yechilgan, ya’ni tizimni bir foydalanuvchi rejimida (bajarilishi darajasi 1 yoki S) qayta yuklash yetarli bo‘lgan. Bu holatda tizim parol so‘rovisiz super foydalanuvchi huquqi bilan komandalar satrini taqdim etadi. Zamonaviy distributivlarda bir foydalanuvchi rejimida yuklash super foydalanuvchining parolini kiritishni talab qiladi (sulogin dasturi yuklanadi). Shuning uchun root foydalanuvchisining parolini almashtirish uchun boshqa metodlarni qo‘llash kerak bo‘ladi.

Root foydalanuvchisining parolini almashtirishning asosiy masalasi dasturlarning bajarilishini super foydalanuvchi huquqidan foydalanish imkonini olishdan iborat. Bunday imkoniyatni olishning ikkita usuli mavjud:

*Birinchi usul* Linux yadrosi parametrlarini bilishni o‘z ichiga oladi. Boshqa dasturlar kabi yadroga yuklanishi uchun komandalar satri parametrlari berilishi mumkin. Parametrlardan biri standart init dasturini boshqa bir ixtiyoriy dasturga almashtirish imkoniyatini beradi. Bizning holda yadroning yuklanishidan keyin darhol bash yoki ixtiyoriy boshqa interpretatorni yuklash kerak. Bunday holda foydalanuvchining autentifikatsiyalash mexanizmini yuklovchi skript bajarilmaydi va biz root foydalanuvchisi huquqi bilan komandalar satriga ega bo‘lamiz.

Ishga tushiruvchi dasturlarga o‘zgartirish kiritish uchun, yadroga quyidagi parametрни ko‘rsatish kerak:

```
init=/dastur/ga/yo`l.
```

Ishga tushgandan keyin, ishga tushirish skripti bajarilmasdan, ildiz katalog faqat o‘qish rejimida o‘rnatiladi. Bizning vazifamiz, parolni almashtirishga urinishdan oldin ildiz katalogni rw rejimida qayta o‘rnatishimiz kerak bo‘ladi. Buning uchun quyidagi komanda ishga tushiriladi:

```
mount -o rw,remount /
```

Keyin root foydalanuvchisining parolini almashtirish uchun passwd dasturini chaqirish yetarli.

```
passwd
```

va albatta diskdagi fayl tizimining buferidan ma’lumotlarni tozalash kerak:

```
sync
```

Keyin kompyuterni qayta yuklash lozim.



*Super foydalanuvchining parolini almashtirishning ushbu usulidan faqat operatsion tizimning yuklovchisiga parol o'rnatilmagan bo'lsagina foydalanish mumkin.*

*Ikkinchi usul*, tashqi ma'lumot tashuvchilar bilan yuklash zarur bo'ladi: yuklovchi disketa yoki CD-ROM orqali. /etc faylida joylashgan fayl tizimiga ulanish kerak. Keyin foydalanuvchining parolini shadow faylida o'zgartirish zarur bo'ladi.

Misol uchun, ildiz katalog /dev/hda3 bo'limida joylashgan. Biz, Slackware Linux distributivini birinchi CD-ROMini yuklaymiz. Ildiz katalog RAM diskida joylashgan. Endi biz tiklamoqchi bo'lgan ildiz tizim joylashgan bo'limga ulanish zarur:

```
mount /dev/hda3 /mnt
```

Bizni qiziqtirayotgan shadow fayli /mnt/etc katalogida joylashgan bo'ladi. Lekin uni qo'lda tahrirlashning hojati yo'q. Biz parollarni o'zgartirish uchun mo'ljallangan standart passwd dasturini chroot nomli boshqa dasturni qo'llagan holda ishga tushiramiz:

```
chroot /mnt passwd
```

Chroot dasturi fayl tizimiga yuklanadigan dasturlarga kirishni cheklaydi.

Bu dastur /mnt-katalogini ildiz fayl tizimi «shekilli» deb hisoblaydi va ushbu ildizning chegarasidan tashqaridan hech narsani «ko'rmaydi». Passwd dasturi /etc/shadow faylidagi parolni o'zgartiradi, lekin /mnt-da joylashgan ildiz fayl tizimini emas, u /mnt/etc/shadow faylining tarkibini o'zgartiradi.

1. Shaxsiy kompyuterda Linux operatsion tizimining yuklanish ketma-ketligi qanday?

2. Linux yadrosi yuklangandan keyin darhol qanday dasturlar yuklanadi?



3. SystemV initsializatsiya tizimi holatida /etc/rc.d katalogida qanday fayl va kataloglar joylashishini yozib chiqing?

4. SystemV initsializatsiya tizimida ishga tushiruvchi skriptlar qo'llab quvvatlaydigan qanday parametrlar

bo'lishi shart?

5. Bajarilishning uchinchi darajasiga o'tishida Slackware Linux initsializatsiya tizimida qanday fayl bajariladi?

6. Slackware Linux-da tizimning ishga tushishida qandaydir ishga tushiruvchi skriptning yuklanishini inkor qilish uchun nima qilish kerak?

## 3-BOB. LINUXDA QURILMALAR BILAN ISHLASH

### 3.1-§. Linuxda qurilmalar bilan ishlash

- Qurilmalar fayli
- Linux yadro modulini yuklash
- Modprobe dasturi konfiguratsiyasi

Linux-da modulni yuklashning uchta usuli qo'llaniladi:

- “Qo‘lda” yuklash usuli, bunda modprobe dasturini oshkor holda komandalar satri yoki start beruvchi skript yordamida chaqirish bilan;
- yadro yuklanishida modulning avtomatik ishga tushirilishi;
- **hotplug** – qaynoq tizimi yordamida yuklash.

Slackware va SuSE Linux-larda yuqoridagi uchta yuklash usulini ham qo'llash mumkin. Linux-ning qolgan asosiy distributivlarda keyingi ikkita yuklash usulidan foydalaniladi.

Biz quyida modulni yuklashning ikkinchi usulini qarab chiqamiz. Birinchi usul Slackware Linux-ning start beruvchi skriptlari rc.modules va rc.netdevise-larda keltirilgan.

**Qurilma fayllari.** Qurilma fayllari quyidagi parametrlar bilan aniqlanadi:

- qurilma turi
- katta nomeri
- kichik nomeri

Qurilma faylini yaratish uchun quyidagi dastur qo'llaniladi:  
mknod nom <b|c> major minor

Yadro modullarini qanday yuklanishini tushinish uchun, qurilma fayllari nima ekanligi, nima uchun ishlatilishi va qanday ishlashi haqida batafsil ma'lumotga ega bo'lish kerak.

Qurilma fayliga murojaat qilishda tizim dasturlarning qurilmaning mos drayveriga murojaat qilishini ta'minlaydi. Dasturchilar qurilma fayllari bilan ishlash uchun tizimning standart funksiyalaridan foydalanishi mumkin, masalan: open, read, write va close. Lekin dasturchi drayverga qanday formatdagi ma'lumotni kiritish yoki undan qanday ma'lumotni o'qishni bilishi shart.



*Qurilma fayllari qurilmaning drayveri emasligini tushinishi shart. Modullar – bu qurilma drayverlari. Qurilma fayli – bu qurilma drayveriga kirish uchun*

*interfeys.*

Agar /dev katologidagi fayllarning atributlariga e'tibor qaratsak, qiziqarli ma'lumotga ega bo'lishimiz mumkin, ya'ni barcha qurilma fayllarining o'lchami o'rniga ikkita raqam joylashtirilgan (bu fayllar birinchi harfi bo'yicha aniqlanadi):

```
brw-rw ---- 1 root disk 3, 0 Apr 26 2016
/dev/hda
brw-rw ---- 1 root disk 3, 1 Apr 26 2016
/dev/hda1
brw-rw ---- 1 root disk 3, 10 Apr 26 2016
/dev/hda10
```

Ixtiyoriy qurilma fayli uchta parametr bilan xarakterlanadi:

- qurilma turi – blokli yoki belgili.
- katta (major) nomeri – drayver turini aniqlaydi.
- kichik (minor) nomeri – aniq fizik qurilmani aniqlaydi.

Bizning misolda uchta fayl ko'rsatilgan: hda, hda1 va hda10. Bular blokli qurilmalar bo'lib, satrning boshidagi "b" belgisi nimani anglatadi. Bu qurilmalarning katta nomeri bir xil – 3 ga teng. Har bir qurilmaning kichik nomeri o'ziniki alohida.

Qurilma fayllarini yaratish uchun mknod dasturi qo'llaniladi.

mknod fayl\_nomi b|c major minor

Agar qaysi qurilmaga qanday nomdagi fayl mos kelishini bilmasak, u holda Linux yadrosining hujjatiga (ma'lumotnomasiga) murojaat qilish kerak. Ya'ni Documentation/devices.txt faylida barcha qurilma fayllarining nomlari va fayllar atributlari to'liq berilgan.

### **3.2-§. Linux yadro modulini yuklash**

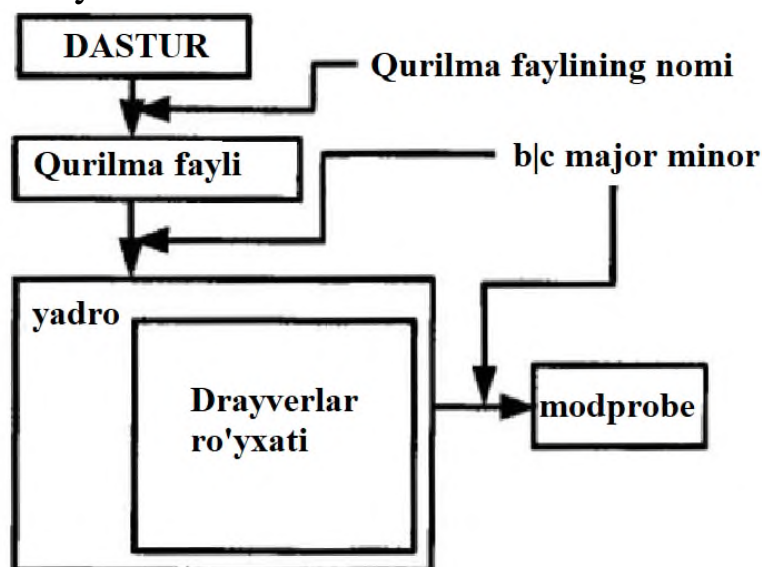
Dastur kodida qurilma fayliga uning nomi bo'yicha murojaat bo'lganda, yadroga fayl nomi haqidagi ma'lumot emas, balki qurilmani tavsiflovchi bir qiymatli uchta parametr beriladi:

- qurilma turi.
- katta nomeri.
- kichik nomeri.

Aynan mana shu xususiyat uchun ham qurilma faylining nomi ixtiyoriy nom bo'lishi mumkin.

Yadro mos parametrlarni olgandan so'ng ayni vaqtda mavjud bo'lgan qurilmalarning drayverlari ro'yxatini qarab chiqadi. Agar u

zarur drayverni topsa, u holda dastur mos kelgan drayver bilan muloqotni boshlaydi.



Agar qurilma drayveri bo'lmasa, u holda yadro /proc/sys/kernel/modprobe faylida tavsiflangan dasturni chaqiradi. Jimlik holatida bu faylda *modprobe* dasturiga yo'l yozilgan bo'ladi. Bu dasturni chaqirishda yuklanishi zarur bo'lgan modul nomi emas, balki yadro olgan parametr beriladi. Modprobe dasturi zarur modulni yuklaydi (agar mumkin bo'lsa). Yadro drayverlari ro'yxatida yangi drayver nomi paydo bo'ladi va dastur u bilan muloqotni boshlaydi.

### 3.3-§. Modprobe dasturining konfiguratsiyasi

Modprobe dasturining konfiguratsiya fayli:  
**/etc/modprobe.conf**

-c parametri jimlik holatida konfiguratsiyani ko'rsatadi.

Oldingi mavzuda aytilganidek, yadro modulni yuklash uchun modprobe dasturini ishga tushiradi. Parametr sifatida yadroga odatda katta va kichik nomeri ko'rsatilgan qurilma turi beriladi. Masalan quyidagicha:

**char=maior-10-144**

Ba'zi turdagi qurilmalar uchun maxsus fayl mavjud emas. Bu turdagi qurilmalarga tarmoq va ovoz kartalari kiradi.

Tijorat (sotiladigan) UNIXdan farqli ravishda Linux ko'p sondagi turli qurilmalarni qo'llab-quvvatlaydi. Shuning uchun ham

Linux-da ko'rsatilgan qurilma uchun oldindan aniqlangan qurilma fayli yo'q.

Masalan, ifconfig dasturi konfiguratsiya uchun tarmoq kartasini chaqirganda, unga qurilma nomi sifatida eth0 beriladi. /dev katologida eth0 fayli yo'q. Shuning uchun ifconfig dasturi yadroga murojaatida qurilma nomiga eth0-ni beradi. Aynan shunday nomni modprobe dasturi yadroga beradi.

Ikkala holda ham (qurilma parametri yoki qurilma nomi berilganda) modprobe dasturiga mos qurilma turi va yuklanadigan modulni ko'rsatish shart.

modprobe dasturida jimlik holatida konfiguratsiya mavjud bo'ladi. Uni «-s» parametri bilan dasturni bajarish orqali ko'rish mumkin. Lekin, agar mos eth0 qurilmasining satri topilsa, bu qurilma uchun modul nomi aniqlanmaganini ko'rish mumkin:

```
alias eth0 off
```

modprobe dasturining konfiguratsiyasini aniqlash yoki oldindan aniqlash uchun uning konfiguratsiya fayli /etc/modprobe.conf da amalga oshiriladi.

Faraz qilaylik, kompyuterga uchta tarmoq kartasi o'rnatilgan bo'lsin: ikkita ISA shinasiga mos NE2000 va PCI shinasiga mos NE2000. Ularga ne va ne2k-pci modullari mos keladi. Tarmoq kartochka qurilmalarini aniqlash uchun /etc/modprobe.conf fayliga quyidagi uchta satrni qo'shish kerak:

```
alias eth0 ne  
alias eth1 ne  
alias eth2 ne2k-pci
```

Ya'ni, alias parametri yordamida mos qurilma va yuklanishi zarur bo'lgan modul aniqlanadi.

ISA shinasidagi NE2000 tarmoq kartasining bitta xususiyati bor – modul yuklanishida ushbu kartalar konfiguratsiya qilingan kiritish-chiqarish portlari ko'rsatilishi zarur.

Modprobe dasturining bevosita chaqirilishida komandalar satrida quyidagicha yozish mumkin:

```
modprobe ne io=0x280, 0x320
```

/etc/modprobe.conf faylini qo'llagan holda komandalar satriga yana bitta satrni qo'shib yozish kerak:

`options ne io=0x280, 0x320`

Ya'ni, modul yuklanishida options yordamida juda kam parametrlari aniqlanadi.

Yuqorida sanab o'tilgan parametrlardan va /etc/modprobe.conf faylidan tashqari boshqa parametrlarni qo'llash mumkin. Bularning ba'zilari quyidagi jadvalda keltirilgan:

<b>Parametr</b>	<b>Tavsifi</b>
pre-install moduli	Ko'rsatilgan modul yuklanishidan oldin bajariladigan komanda
post-install modul, komanda	Ko'rsatilgan modul yuklangandan keyin bajariladigan komandani aniqlash
pre-remove modul, komanda	Modulni o'chirishdan oldin bajarilishi aniqlanadigan komanda
post-remove modul, komanda	Modul o'chirilganlan keyin bajariladigan komandani aniqlash

1. Tizimdagi qurilmalar fayli nima uchun kerak?

2. /dev/hdb interfeysiga mos keluvchi qurilma faylini yaratish dasturi mkndod-ga berilishi zarur bo'lgan parametrlarni qaerdan qidirish kerak?



3. /dev/hdb qurilmasiga mos keluvchi o'zining parametrlari (mojor, minor)ga ega bo'lgan /dev/myBigDisk nomli qurilma faylini yaratish mumkinmi?



### 3.4-§. ldconfig va ldd dasturlari

Linux-da dinamik yuklanuvchi kutubxonalar qo'llaniladi. Dastur shunday kutubxonada joylashgan funksiyani qo'llashi mumkin, u fayl tizimida joylashgan kutubxonalar fayli qayerda ekanligini bilishi shart, chunki bu fayl uni tezkor xotiraga yuklash uchun zarur bo'ladi.

Lekin tizim administratori kutubxonalar fayllarini ixtiyoriy katalogga joylashtirishi mumkin. Dasturni yozayotgan dasturchi bunday fayllarni qandaydir usulda qайдan qidirishni bilishi shart. Bu muammoni yechish uchun **Linux-da** kesh kutubxonasi deb ataluvchi kutubxonalar qo'llaniladi. Bu kutubxonalar binar ma'lumotlar bazasi deb ataluvchi /etc/ld.so.cache faylida saqlanadi.

Keshni yaratish uchun ldconfig dasturi qo'llaniladi. Bu dastur /etc/ld.so.conf faylida aniqlangan barcha kataloglarni qarab chiqadi va unda topilgan barcha kutubxonani keshga joylashtiradi.

Quyidagi /etc/ld.so.conf faylining mazmuni keltirilgan:

```
/usr/local/lib
/usr/x11R6/lib
/usr/i486-slackware-linux/lib
/opt/kde/lib
/usr/lab/qt/lib
```



*Bu faylda /lib va /usr/lib standart kataloglar yozilganligiga e'tibor bering. Xuddi shunday bu faylda faqat yuqori darajali kataloglarni yozish yetarli bo'lib, qism kataloglarni ldconfig dasturining o'zi tekshirib chiqadi.*

Keshning tarkibini **ldconfig** dasturining o'zi bilan "-r" parametrini ko'rsatish orqali ko'rish mumkin.

```
# ldconfig -p | head -5
1073 libe found in cache ` /etc/ld.so.cache `
  libx.so.1 (libc6) => /usr/lib/libx.so.1
  libx.so (libc6) => /usr/lib/libx.so
libyafxplayer.so (libc6) => /opt/kde
/libyafxplayer.so
libyafcore.so (libc6) => /opt/kde/lib
/libyafcore.so
#
```

Yangi kutubxonani qo‘shgandan keyin albatta **ldconfig** dasturini qayta yuklash kerak. Chunki, u yangi keshni tayyorlaydi va dastur o‘rnatilgan kutubxonani ko‘rish imkoniyati paydo bo‘lishi kerak.

Agar yangi kutubxona nostandart katalogda joylashtirilgan bo‘lsa, bu katalog nomi **/etc/ld.so.conf** fayliga qo‘shilgan bo‘lishi shart. Shundan keyin **ldconfig** dasturini albatta qayta yuklash kerak bo‘ladi.

**ldd** dasturi yordamida dasturning ishlashi uchun zarur bo‘lgan qanday kutubxonalar borligini ko‘rish mumkin.

```
# ldd /usr/bin/perl
linux-gate.so.1 => (0xffffe000)
libnsl.so.1=>/lib/tls/libnsl.so.1(0xb7f6e000)
libdl.so.2=>/lib/tls/libdl.so.2(0xb7f6a000)
libm.so.6=>/lib/tls/libm.so.6(0xb7f47000)
libcrypt.so.1=>/lib/tls/libcrypt.so.1(0xb7f19000)
libutil.so.1=>/lib/tls/libutil.so.1(0xb7f15000)
libc.so.6 => /lib/tls/libc.so.6(0xb7de6000)
/lib/ld-linux.so.2(0xb7f9b000)
#
```

Dasturni **chroot** dasturida joylashtirishda qo‘llash juda qulay

### 3.5-§. IDE kontrollerni sozlash

Linux-da IDE interfeysni sozlash imkoniyati mavjud. Bu sozlama ba’zi qurilmalar jimlik holatidagi sozlamalar bilan yaxshi ishlamagan yoki umuman ishlamagan hollarda zarur bo‘ladi.



*Interfeysga o‘zgartirish kiritish qurilmaga zarar keltirishi mumkin. Shuning uchun o‘zgartirish kiritishdan oldin qurilma dasturini va hujjatlarini diqqat bilan o‘qib chiqish kerak bo‘ladi.*

#### **hdparm dasturi.**

**hdparm [parametrlar][qurilma]**

Linux-da IDE interfeysni sozlash uchun **hdparm** dasturi qo‘llaniladi.

Bu dastur yordamida quyidagilarni amalga oshirish mumkin:

- kontrollerga ulangan qurilma kontrolleri joriy parametrlarini

ko'rish.

- **IDE** kontrollerning parametrlarini o'zgartirish.
- qurilmaning tezligini tekshirish.

Quyidagi jadvalda hdparm dasturining ba'zi parametrlari keltirilgan:

<b>Parametr</b>	<b>Tavsifi</b>
-dN	Agar N: 0 ga teng bo'lsa – DMA ning qo'llab-quvvatlashi o'chiradi 1 ga teng bo'lsa– DMA ning qo'llab-quvvatlashi yoqiladi.
-cN	Agar N: 0 ga teng bo'lsa – 16 bitli kiritish-chiqarish rejimi qo'shiladi 1 ga teng bo'lsa – 32 bitli kiritish-chiqarish rejimi qo'shiladi 3 ga teng bo'lsa – 32 bitli kiritish-chiqarish rejimi sync ketma-ketligini qo'llash bilan qo'shiladi.
-mN	Agar N: 0 ga teng bo'lsa – qattiq disk shpindelining bir marta aylanishida bir nechta sektorlarni o'qish imkoniyati o'chiriladi. Son bo'lsa – shpindelning bir marta aylanishida bir vaqtda N ta sektorni o'qish imkoniyati o'rnatiladi.
-EN	Bu parametr CD-ROM ning tezligini o'rnatadi
-f	Buferdagi ma'lumotlarni qattiq diskka o'tkazadi, bufer tozalanadi.
-g	Qattiq diskning geometrik shaklini ko'rsatadi
-i	Tizim yuklanish vaqtida olingan qurilmalar haqidagi ma'lumotlarni ko'rsatadi
-I	Qurilma haqidagi ma'lumotni so'raydi. -i parametriga nisbatan ko'proq ma'lumotlarni chiqaradi.
-T	Tezkorlikni tekshirish uchun mo'ljallangan parametr. Buferdan o'qish tezligini ko'rsatadi.
-t	Tezkorlikni tekshirish uchun mo'ljallangan parametr. Qurilmadan o'qish tezligini ko'rsatadi.

-X	DMA rejimini o'rnatadi
----	------------------------

Birinchi o'rinda kotroller haqidagi ma'lumotni olish kerak. Bu quyidagi komanda yordamida amalga oshiriladi:

hdparm/dev/hda



*Agar DMA qo'llab quvvatlashi o'chirilgan bo'lsa, tizim katta hajmdagi fayllar ustida amallar bajarishda juda sekin ishlaydi. Imkoni boricha DMA rejimi yoqilgan bo'lishiga e'tibor berish kerak.*

Qurilma haqida to'liq ma'lumot olish uchun -I parametri qo'llaniladi:

hdparm -I /dev/hda

Qurilmaning tezkorligini tekshirish -t va -T parametrlari orqali amalga oshiriladi:

hdparm -tT /dev/hda



*Qurilmaning tezkorligini tekshirish vaqtida tizimda boshqa jarayon qo'llanilayotgan bo'lishi mumkin, shuning uchun yuqorida ko'rsatilgan komandani bir necha marta qayta takrorlab, tezkorlikning o'rtacha qiymatini olish kerak bo'ladi.*

Qurilma uchun qo'llaniladigan ish rejimi haqida ma'lumotlarni olgandan keyin, unga maksimal qiymatni o'rnatish mumkin. Asosiy e'tiborni DMA rejimini o'chirishga, kiritish-chiqarishda bir vaqtda o'qish mumkin bo'lgan sektorlar soniga qaratish zarur (MaxMultiSec).

Faraz qilaylik **IDE** interfeysi sync (sinxronlash) ketma-ketligini qo'llovchi 32 bitli rejimni qo'llab quvvatlaydi, UDMA5 va bir vaqtda o'qishi mumkin bo'lgan sektorlarning maksimal soni 16 teng bo'lsin. U holda ushbu rejimni qo'shish uchun dasturni ishga tushirish quyidagi namuna asosida amalga oshiriladi:

**hdparm -d1 -c3 -m16 -X udma5 /dev/hda**

yoki

**hdparm -d1 -c3 -m16 -X64 /dev/hda**

Yuqorida keltirilgan satrlar bir biridan faqat -X parametrni kiritilishi bilan farq qiladi. -X parametri yordamida UDMA rejimi aniqlanadi. Bu rejimni ikkita usulda ko'rsatish mumkin:

**-X udma5**

Bu usulda udma5 satri qo'llaniladi. Shuni alohida ta'kidlash kerakki, parametr va uning qiymati orasida albatta probel qo'yilishi shart.

Parametrning ikkinchi variantda berilishi quyidagicha bo'ladi:  
**-x64**

Bu variantda parametrdan keyin yozilgan son probelsiz qo'yiladi. Bu son UDMA rejimini 64 nomerini qo'shish orqali oladi.

### **3.6-§. Tizimning dastlabki yuklanishida qattiq disk parametrlarini sozlash**

Tizim qayta ishga tushirilganda, IDE kontrollerining sozlamalari tiklanmaganligi sababli, hdparm dasturini tizimni ishga tushirishda kerakli parametrlar bilan bajarilishini ta'minlash kerak.

Bunga ikkita usul yordamida erishish mumkin:

- /etc/rc.d/rc.local faylida dasturni chaqirishni tavsiflash;
- distributivning standart o'rnatilishini qo'llash.

Dastlab RedHat Linux distributivida buni qanday qo'llash mumkinligi ko'rib chiqamiz.

RedHat Linux tizimida hdparm dasturini chaqirish /etc/rc.d/rc.sysinit faylida paydo bo'ladi. Ya'ni tizimning yuklanishining ixtiyoriy bosqichida bajariladi.

Hdparm dasturiga /etc/rc.d/rc.sysinit skriptida parametrlarni berishda quyidagi fayllarni qo'llash mumkin:

- /etc/sysconfig/harddisks – barcha kontrollerlarga umumiy parametrlarni uzatish uchun
- /etc/sysconfig/harddiskshd[a-h] – aniq qurilmaga parametrlarni uzatish uchun.

Quyidagi jadvalda ko'rsatib o'tilgan fayllarda belgilanishi lozim bo'lgan parametrlar ro'yxati berilgan:

<b>Parametrlar</b>	<b>Tavsifi</b>
USE_DMA	Bu o'zgaruvchining qiymati –d parametriga qo'yiladi. Agar parametr qiymati 1 bo'lsa, DMA qo'llaniladi, aks holda 0 bo'lsa, o'chiriladi.
MULTIPLE_IO	Bu o'zgaruvchining qiymati –m parametriga

	qo'yiladi, agar bu qiymat 0 bo'lsa, qattiq disk shpindelining bir marta aylanishida bir nechta sektorlarni o'qish rejimi qo'llaniladi. Boshqa son qo'yilsa, bu son bir marta aylanishda nechta sektor bir vaqtda o'qilishini anglatadi.
EIDE_32BIT	Bu o'zgaruvchi qiymati -s parametriga qo'yiladi. 0 - kiritish-chiqarishning 16 bitli rejimi, 1 - 32 bitli rejim, 3 - sync ketma-ketligini qo'llovchi 32 bitli rejim aniqlanadi.
LOOKAHEAD	Bu o'zgaruvchi qiymati -A parametriga qo'yiladi. Agar qiymat 0 bo'lsa, - ma'lumotni oldindan o'qish rejimi o'chiriladi, 1 - yoqiladi.
EXTRA_FARAMS	Bu o'zgaruvchiga maxsus o'zgaruvchilar tayinlanmagan bo'lsa hdparm dasturining qo'shimcha parametrlarini yozish kerak.

Oldingi mavzuda berilgan interfeysni sozlash misoliga qaytadigan bo'lsak, /etc/sysconfig/harddiskshda faylini quyidagi mazmunda hosil qilishimiz kerak:

```
USE_DMA=1
MULTIPLE_IO=16
KIDE_32BIT=3
EXTRA_PARAMS= "-X udma5"
```

SuSE Linux-da boshqa konfiguratsiya fayli qo'llaniladi: /etc/sysconfig/ide. Unda faqat bitta o'zgaruvchi mavjud - DEVICES\_FORCE\_IDE\_DMA.

Yuqorida tavsiflangan sozlashni qo'shish uchun, bu o'zgaruvchiga quyidagi qiymatni kiritish kerak bo'ladi::

```
DEVICES_FORCE_IDE_DMA= "/dev/hda:udma5:-m16:-c3"
```

Masalan, agar qo'shimcha ravishda /dev/hdb interfeysiga DMA ni qo'llab quvvatlashni o'chirish kerak bo'lsa, u holda o'zgaruvchini aniqlash quyidagi ko'rinishda bo'ladi:

```
DEVICES_FORCE_IDE_DMA=          "/dev/hda:udma5:-m16:-c3
/dev/hdb:off"
```

Qolgan interfeyslarning parametrlari ham probel orqali namunadagi kabi yoziladi.

Slackware Linux-da ishga tushiruvchi skriptda qo'shimcha

konfiguratsiya fayli mavjud emas. Lekin skriptning o'zini o'zgartirish imkoniyati bor. Shuning uchun /etc/rc.d/rc.S faylida zarur parametrlarga ega hdparm dasturini chaqirish yetarli bo'ladi. Bu tizimning dastlabki yuklanishini bajaradi va shuningdek, IDE interfeysining zarur ish rejimida o'rnatilishini ta'minlaydi.

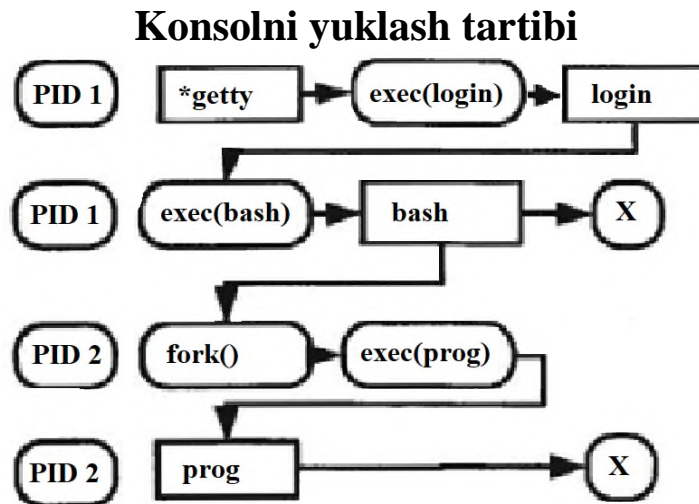
1. hdparm dasturida -X64 parametri nimani anglatadi?
2. /dev/hdd interfeysiga ulangan CD-ROM-ni DMA-ni qo'llashini qanday o'chirib qo'yish mumkin?
3. /dev/hdd interfeysiga ulangan CD-ROM-da tizimni ishga tushirgandan so'ng DMA-rejimini o'chirish uchun qanday faylni yaratish kerak. SuSE Linux va uning tarkibida ishlatiladigan yuklovchi tizimining xususiyatlarini hisobga olgan holda fayl nomini yozing.



### 3.7-§. Foydalanuvchi loginini sozlash

Linux administratori foydalanuvchilarning tizimga kirishini boshqara olish bilimlariga ega bo'lishi shart. Bu mavzuda quyidagilarni o'rganamiz:

- konsolni yuklash tartibi.
- login dasturini qanday sozlash.
- foydalanuvchilarning tizim resurslariga kirish ruxsatini qanday cheklash.



Init dasturining konfiguratsiya fayli - /etc/inittab da \*getty<sup>5</sup> dasturini yuklash satri mavjud emas. Bu dasturning asosiy vazifasi, konsolni ketma-ket qurilmada yuklashdan iborat. Odatda init dasturi tty1-dan tty6-gacha bo'lgan virtual konsol-lar uchun oltita nusxasi yuklanadi. Dastur ishining tugashidan keyin init yangidan \*getty dasturini ishga tushiradi, chunki inittab faylining action maydonida respawn holati yozilgan.

Konsol yuklangandan keyin \*getty dasturi login dasturini ishga tushiradi. login, password so'rovi kiritilgandan keyin, foydalanuvchini identifikatsiya qiladi.

Agar foydalanuvchi /etc/passwd faylining yettinchi maydonida tavsiflangan login va parolni to'g'ri kiritrsa, dastur ishga tushadi. Odatda u yerda bash qobig'i yozilgan.

Xabarlarning yetarlicha uzun zanjiri olinadi. Shunga qaramasdan, agar init dasturida yuklangan dastur o'z ishini tugatgan bo'lsa, u yana qaytadan yuklanadi. Dasturning yuklanish zanjiriga yana bir marta e'tibor bersak, \*getty dasturi login dasturining

<sup>5</sup> Dastur nomi distributiv turiga bog'liq



yuklanishidan oldin yuklanishi kerak. Lekin aslida bu bajarilmaydi. \*getty dasturi bash qobig'ining yuklanishi bajarilgandan keyin ishga tushadi.

Gap shundaki, \*getty va login keyingi dasturni ishga tushirish uchun yangi jarayonni hosil qilmasdan, joriy holatni o'zgartiruvchi yexes tizimli funksiyasidan foydalanadi. Ya'ni, exec (login) dan so'ng, login dasturi \*getty dasturida bo'lgani kabi bir xil PID bilan amalga oshiriladi. init esa yuklanadigan dasturning PID-ini qidiradi. bash qobig'ining ishi tugashi bilan PID tizimga qaytadi. Aynan shundan keyin init dasturi \*getty ni qayta yuklaydi.

Dasturlarni bash da yuklash uchun boshqa mexanizm qo'llaniladi. Dastlab bash fork funksiyasi yordamida bo'sh turgan PID ga ta'minlanadigan yangi jarayonni hosil qiladi va dasturni yuklash uchun ushbu yangi jarayon yexes funksiyasini chaqiradi.

Yuqorida sanab o'tilgan dasturlardan faqat login dasturining konfiguratsiyasini ko'rib chiqamiz. bash konfiguratsiyasi nazariy va amaliy mashg'ulotlarda ko'rib chiqilgan. \*getty esa kamdan-kam hollarda sozlanadi.

### 3.8-§. Login dasturining konfiguratsiyasi.



*E'tibor bering. login dasturi konfiguratsiyasiga kiritiladigan foydalanuvchiga barcha cheklovlar, faqat lokal konsolga kirish uchun qo'llaniladi, tarmoqqa kirish cheklovlari telnet dasturi yordamida amalga oshiriladi. Foydalanuvchining tarmoqqa kirishni ssh dasturi yordamida sozlash sshd xizmati konfiguratsiyasini sozlash orqali amalga oshiriladi.*

Alohida ta'kidlash kerakki, Linuxning turli distributivlarida dasturlarning turlicha konfiguratsiyasidan foydalaniladi. Slackware **Linux-da** bu fayllar quyidagicha bo'ladi:

- /etc/login.defs

- /etc/login.access

SuSE va RedHat **Linux-da** esa:

- /etc/login.defs

- RAM modelining konfiguratsiya fayllari:

- /etc/secure/access-conf

- /etc/nologin

- /etc/securetty
- va boshq.

Bu shuni anglatadiki, Linux-ning oxirgi distributivlarida RAM moduli qo'llaniladi, Slackware Linux-da esa bu qo'llanilmaydi. Shu sababli Slackware **Linux-dagi** /etc/login.defs faylining mazmuni boshqa distributivlardagi o'xshash fayllarga nisbatan ko'proq parametrlar qiymati mavjud.

**/etc/login.defs fayli.** Bu fayl nafaqat login dasturining konfiguratsiyasi, balki quyidagi dasturlar uchun ham konfiguratsiya fayli hisoblanadi:

- passwd
- useradd
- userdel
- chfn
- pwconv
- groupadd
- su

Quyidagi **login.defs** faylida qo'llash mumkin bo'lgan ba'zi parametrlar keltirilgan:

Parametrlar	Tavsifi
CONSOLE	Tizimga root foydalanuvchi sifatida kirish konsolini tavsiflovchi faylni aniqlaydi (odatda /etc/securetty fayli). Parametrlarni qo'llashning boshqa varianti - login.defs faylining o'zida ikki nuqta orqali konsolni tavsiflashdir. SuSE va RedHat Linux-da esa bu fayllar RAM moduli yordamida aniqlanadi.
MOTD_FILE	Foydalanuvchining tizimga kirishida mazmuni kiritiladigan fayl nomini aniqlaydi. Slackware Linux-da tizimning har bir yuklanishida bu faylda joriy versiyasi joylashadi. Agar xabarning o'zgarishini xoxlamasak, ikki nuqta orqali ikkinchi faylni aniqlang va unga xabarni joylashtiring.
ISSUE_FILE	Tizimga kirish so'rovi chiqarilishidan

	oldin chiqadigan faylning mazmunini aniqlaydi.
NOLOGINS_FILE	Agar faylda bu parametr aniqlangan (kiritilgan) bo'lsa – tizimga superfoydalanuvchidan boshqa foydalanuvchilar kira olmaydi. Faylning mazmuni foydalanuvchi tizimga kirishga urinishni amalga oshirganda ekranga chiqadi. SuSE va RedHat Linux-da esa bu fayllar RAM moduli yordamida aniqlanadi.
QMAIL_DIR MAIL_DIR MAIL_FILE	Foydalanuvchining pochta qerda va qanday ko'rinishda saqlanishini aniqlaydi. Barcha distributivlarda majburiy aniqlanadigan parametrlar toifasiga kiradi
ENV_SUPATH ENV_PATH	Tizimning oddiy va super foydalanuvchilari uchun PATH o'zgaruvchisiga dastlabki qiymatlarni aniqlash parametri. Barcha distributivlarda majburiy aniqlanadigan parametrlar toifasiga kiradi
PASS_MAX_DAYS	Parolning hayotiy tsikli uchun maksimal vaqt. Useradd dasturi uchun jimlik holatidagi qiymat.
PASS_MIN_DAYS	Foydalanuvchi parolni o'zgartirish huquqiga ega bo'lmaganda, parolni oxirgi o'zgartirgandan keyingi kunlar soni. Useradd dasturi uchun jimlik holatidagi qiymat.
PASS_MIN_LEN	Passwd dasturi foydalanuvchidan qabul qilishi mumkin bo'lgan parolning minimal uzunligi. Useradd dasturi uchun jimlik holatidagi qiymat
PASS_WARN_AGE	Foydalanuvchining paroli yaroqlilik muddatiga necha kun qolganligi haqida tizimga kirishda ogohlantiruvchi xabar chiqariladi. Useradd dasturi uchun jimlik holatidagi qiymat
UID_MIN	Useradd dasturida qo'llaniladigan UID-

UID_MAX	ning maksimal va minimal qiymatlarini aniqlaydi, yaratilayotgan foydalanuvchiga UID avtomatik biriktiriladi.
GID_MIN GID_MAX	Bu ham yuqoridagidek, faqat groupadd dasturi va GID uchun
SU_WHEEU_ONLY	Agar bu parametrning qiymati «Yes» ga teng bo'lsa, su dasturini faqat root guruhiga kiruvchi foydalanuvchilar qo'llashi mumkin bo'ladi.

### 3.9-§. /etc/login.access va /etc/secure/access-conf fayllari

Slackware Linux-da /etc/login.access fayli qo'llaniladi, boshqa distributivlarda esa /etc/secure/access.conf faylidan foydalanamiz.

Bu fayllar yordamida tizimga qaysi foydalanuvchi va qaerdan kirishi mumkin yoki kirishi mumkin emasligini ko'rsatish mumkin. Jimlik holatida bu faylda yozuvlar bo'lmaydi. Shuning uchun dastlab foydalanuvchilarning tizimga kirishida cheklovlar bo'lmaydi.

Fayl satrlarining formati:

permission : users : origins

- permission

+ - kirishga ruxsat, - - kirish taqiqlangan.

- users – foydalanuvchilar ro'yxati.

- origin – tizimga foydalanuvchi qaerdan kirishga urinishi mumkin.

Bu faylda quyidagi kalit so'zlardan foydalanish mumkin:

- ALL – ixtiyoriy foydalanuvchi yoki manzil.

- EXCEPT – istisno tariqasida sanab o'tilgan foydalanuvchilar yoki manzillar.

- LOCAL- lokal kompyuter.

Masalan, root foydalanuvchisini lokal konsolda kirishini taqiqlash uchun, faylga quyidagi satrlarni qo'shish kerak:

- : root : LOCAL

### 3.10-§. RAM moduli va uning konfiguratsiya fayli.

Ko'plab distributivlarda RAM (Pluggable Authentication Module) moduli texnologiyasidan foydalanish yo'lga qo'yilgan. Tizim modullari yordamida, nafaqat tizimning ichiga, balki autentifikatsiya

tizimiga turli xil funksiyalarni qo'shish mumkin. Buning uchun: dastur RAM ni qo'llab-quvvatlovchi rejimida to'ldirilgan bo'lishi kerak, shundagina RAM modulining barcha imkoniyatlaridan foydalanish mumkin. Slackware Linux-da RAM moduli ishlatilmaydi, bu distributivda nima uchun ishlamasligi haqida keyinroq so'z yuritamiz.

RedHat va SuSE Linux-da login dasturi RAM modulini qo'llab-quvvatlaydi. Bunda administrator login dasturi bilan ishlashda RAM moduliga mos keluvchi ba'zi xususiyatlarni qo'shish yoki o'chirib tashlashi mumkin.

Quyida Red Hat Linux tizimini o'rnatgandan keyin jimlik holatidagi /etc/pam.d/login faylining mazmuni keltirilgan. Qulaylik uchun fayldagi ma'lumotlar nomerlangan satrlar bilan berilgan:

```
1 #%PAM-1.0
2 auth required pam_securetty.so
3 auth required pam_stack.so
service=system-auth
4 auth required pam_nologin.so
5 account required pam_stack.so
service=system-auth
6 password required pam_stack.so
service=system-auth
7 session required pam_stack.so
service=system-auth
8 session required pam_mktemp.so
9 session optional pam_console.so
```

2-satrdan pam\_securetty.so moduli qo'llanilgan bo'lib, bu autentifikatsiya vaqtida /etc/securetty fayliga kirishni ta'minlaydi. Bu faylning qiymatlari keyinroq keltirib o'tiladi.

Uchinchi satrdan system-auth faylining barcha modullari qo'shilgan.

To'rtinchi satrdan pam\_nologin.so moduli qo'llanilgan bo'lib, bunda /etc/nologin faylining vazifalari qo'shilgan. Bu fayl haqida keyinroq keltirib o'tamiz.

login fayliga qo'shilgan system-auth faylining mazmunini ko'rib chiqamiz:

```
1 #%PAM-1.0
2 # this file is auto-generated.
```

```

3 # User changes will be destroyed the next
time authconfig is run.
4 auth required /lib/securetty/$ISA/pam_env.so
5 auth sufficient
/lib/securetty/$ISA/pam_unix.so likeauth nullok
6 auth required /lib/securetty/$ISA/pam_deny.so
7 account required
/lib/securetty/$ISA/pam_unix.so
8 password required
/lib/securetty/$ISA/pam_passwd.so
min=disabled,24,12,8,7 max=40 passphrase=3
match=4 similar=deny random=42 retry=3
onforce=users
9 password sufficient
/lib/securetty/$ISA/pam_unix.so nullok
use_suthtok md5 shadow
10 password required
/lib/securetty/$ISA/pam_deny.so
11 session required
/lib/securetty/$ISA/pam_limits.so
12 session required
/lib/securetty/$ISA/pam_unix.so

```

Bu faylda bizni ikkita satr qiziqtiradi. Birinchisi bu 4-satr. Unda pam\_env.so moduli qo'llanilgan. Bu modul yordamida tizimga kirayotgan foydalanuvchi uchun tashqi o'zgaruvchiga boshlang'ich qiymatni berish mumkin. Modulning konfiguratsiya fayli quyidagicha:

**/etc/environment**

Shuningdek, bizga pam\_limits.so modulini chaqiruvchi 11-satr ham qiziqtiradi. Bu modul yordamida foydalanuvchi uchun tizim resurslarini qo'llashga cheklovlarni aniqlash mumkin. Foydalanuvchi uchun cheklovlar konfiguratsiyasi bilan keyingi bo'limda tanishamiz.

1. Lokal konsol orqali foydalanuvchi kirishiga taqiq qo'yish kerak. Bu qaysi faylning qaysi satriga yoziladi?
2. login dasturining qaysi konfiguratsiya faylida useradd dasturining foydalanuvchi katalogini yaratishga sabab bo'ladigan parametrini aniqlash kerak bo'ladi?
3. Qanday faylning mavjudligi oddiy foydalanuvchilarning tizimga kirishini taqiqlaydi?



### 3.11-§. Disk maydonidan foydalanishni cheklash.

Disk maydonini ishlatish bo'yicha foydalanuvchiga cheklovlar qo'yish Linux yadrosiga o'rnatilgan kvota ajratish tizimi yordamida amalga oshiriladi.

Zamonaviy distributivlarda bu tizimning ikkinchi versiyasi (quota\_v2 moduli) qo'llaniladi, shuning uchun ham kvotalash tizimini dastlabki sozlashda ushbu modulni qo'lda yuklash kerak.

#### **modprobe quota\_v2**

Kvota sozlamasidan keyin tizimning qayta yuklanishida bu modul avtomatik yuklanadi.

Linux-da kvotaga faqat disk bo'limini to'liq birlashtirish mumkin. Alohida katalogga kvota ajratish taqiqlanadi. Shuning uchun ham birinchi o'rinda qaysi bo'limga cheklov qo'yilishini belgilab olish kerak bo'ladi.

Bu fayl tizimlariga bog'lanish momentida ulash parametrlari yordamida amalga oshiriladi:

- usrquota – foydalanuvchi kvotasini belgilash uchun
- grpquota – guruh bo'yicha kvotani belgilash uchun.

Bu parametrlar qanday cheklovlar qo'yilishiga qarab alohida yoki birgalikda ishlatilishi mumkin.

Fayl tizimlarini ushbu parametrlar bilan bog'lash uchun tizimni ishga tushirishda ularning parametrlarini /etc/fstab fayliga qo'shish kerak. Qaysi fayl tizimiga cheklov qo'yish zarur bo'lsa, ushbu tizimga qo'shish kerak bo'ladi. Qolgan fayl tizimlariga teginmaslik kerak.

Faraz qilaylik, foydalanuvchining uy katalogiga kvota belgilash kerak bo'lsin. Buning uchun uy katalogini alohida bo'limga o'tkazish kerak va bu bo'lim uchun mos ulash parametrlarini qo'shish zarur bo'ladi. Masalan, /dev/hda2 foydalanuvchining uy katalogida joylashgan bo'lsin, u holda fstab faylining satrlari quyidagicha ko'rinishda bo'ladi:

```
/dev/hda2 /home ext3 defaults, usrquota, grpquota 0 1
```

**aquota.user va aquota.group fayllari.** Kvotalanishi zarur bo'lgan har bir bo'limning o'zagiga foydalanuvchi va guruh kvotasi haqida ma'lumotlarni saqlab turuvchi ikkita faylni hosil qilish kerak bo'ladi. Berilgan misolda bu fayllarni /home katalogida yaratish kerak.

```
touch /home/{aquota.user, aquota.group}
chmod 600 /home/{aquota.usr, aquota.group}
```

Fayllar root foydalanuvchisida joylashtirilishi kerak va ularga kirish huquqi 600 bo‘ladi.

### 3.12-§. Quotacheck dasturi

Kvotalash tizimini yuklashdan oldin bu fayllarni fayl tizimining joriy kvota holati haqidagi ma’lumotlar bilan to‘ldirish kerak bo‘ladi. Bu quotacheck dasturi yordamida amalga oshiriladi.

Bu dastur nafaqat ma’lumotlarni saqlash uchun, balki kompyuterni noto‘g‘ri o‘chirish holatida ma’lumotlarni tiklash uchun ham qo‘llaniladi. Shuning uchun tizimga start berishda ishga tushiruvchi skriptlarning birida albatta chaqiriladi. (V *Slackware Linux-da* - /etc/rc.d/rc.M).

```
/sbin/quotacheck -avugm
```

Bu misolda qo‘llanilgan parametrlar:

-a – tarmoq fayl tizimidan tashqari /etc/fstab fayliga yozilgan barcha fayl tizimlarini tekshiradi.

-v – ekranga qo‘shimcha ma’lumotlarni chiqaradi.

-u – faqat /etc/fstab fayliga usrquota yozgan ulash parametrlari bilan fayl tizimini tekshiradi.

-g - faqat /etc/fstab fayliga grpquota yozgan ulash parametrlari bilan fayl tizimini tekshiradi.

-m – faqat o‘qish rejimida ishlovchi fayl tizimlarini bog‘lamaslik.

Xuddi shunday aniq fayl tizimini ko‘rsatish mumkin.

### 3.13-§. Quntaon dasturi

Kvotani tekshirish va tiklashdan keyin kvotalash tizimini ishga tushirish mumkin. Bu quotaon dasturi yordamida amalga oshiriladi.

Odatda bu dastur ishga tushiruvchi skriptlardan birida tizimga start berilgan vaqtda yuklanadi. ((V *Slackware Linux-da* - /etc/rc.d/rc.M)).

```
/sbin/quotaon -avug
```

Bu misolda qo‘llanilgan parametrlar:



-a – tarmoq fayl tizimidan tashqari /etc/fstab fayliga yozilgan barcha fayl tizimlarini va avtomatik qo‘shilmaydigan (noauto parametri) fayl tizimlarini qo‘shadi.

-v – ekranga qo‘shimcha ma’lumotlarni chiqaradi.

-u – foydalanuvchi kvotasini qo‘shadi.

-g – guruh kvotalarini qo‘shadi.

Xuddi shunday, dasturni chaqirishda kvota qo‘shilishi kerak bo‘lgan aniq fayl tizimini ko‘rsatish mumkin.

**/sbin/quotaon /dev/hda2**

### 3.14-§. Quotaoff dasturi

Fayl tizimlari uchun kvota mexanizmini o‘chirish uchun qo‘llaniladigan dastur.

Foydalanuvchi kvotasini sozlash. Kvotalash mexanizmi qo‘shilgandan keyin foydalanuvchi uchun kvotalarni sozlash zarur bo‘ladi. Kvota haqidagi ma’lumotlar aquota.user va acuota.group fayllariga yoziladi. Lekin bu fayllar binar ma’lumotlar bazasi hisoblanadi, uning mazmunini o‘zgartirish uchun maxsus edquota dasturidan foydalaniladi.

**edquota -u user**

**edquota -g users**

Birinchi holatda user foydalanuvchisining kvotalarini o‘zgartiradi. Ikkinchi holatda esa users guruhining kvotalarini o‘zgartiradi.

Kvotani tahrirlash uchun dastur vaqtinchalik faylni hosil qiladi va uni vi redaktorida ochadi. Redaktor sifatida boshqa dasturdan foydalanish ham mumkin. Buning uchun VISUAL muhitining ichki o‘zgaruvchilarini aniqlash zarur bo‘ladi. Misol uchun, o‘rnatilgan mc redaktoridan foydalanish uchun o‘zgaruvchilar quyidagi ko‘rinishda aniqlanadi:

**export VISUAL=/usr/bin/mcedit**

Ochiladigan (yaratiladigan) faylda kvota qo‘llaniladigan har bir bo‘limda guruh yoki foydalanuvchi kvotasi haqidagi ma’lumotlar ko‘rsatiladi.

Linux-da foydalaniladigan disk maydoni soni va/yoki yaratilishi mumkin bo‘lgan fayllar sonini cheklash mumkin.

```

Disk quotes for group students (gid 1000):
Filesystem blocks soft hard inodes soft
hard
/dev/hda7 10022 0 100000 594 0 0

```

Ochiladigan faylda har bir bo‘lim uchun ikki guruhdagi sonlar tasvirlanadi: blocks va inodes. Birinchi bo‘lim hajmni aniqlaydi – bitta blok bir Kbaytga teng. Ikkinchi bo‘lim fayllar soni – bitta inode bitta faylga mos keladi.

blocks va inodes ustunlarida joriy holatlar yozilgan va ularni tahrirlash kerak emas. soft va hard ustunlaridagi ma’lumotlarni o‘zgartirish mumkin.

Ikkita chegara mavjud: yumshoq (soft) va qattiq (hard). Foydalanuvchilar birinchi chegaradan oshib ketishi mumkin, ikkinchi chegaradan hech qachon oshmaydi. Agar foydalanuvchi yumshoq kvotadan o‘tib ketsa, unga ogohlantirish beriladi. Agar u hafta davomida ruxsat berilgan chegaraga qaytmasa, u jazolanadi va qattiq kvota yumshoq chegaragacha tushiriladi. Va bu foydalanuvchi yumshoq kvota chegarasiga qaytgandan keyin ham bir hafta davom etadi. Buning uchun foydalanuvchi fayllari o‘chirilmaydi.

Kvota maydonidagi nol kvota yo‘qligini bildiradi. Bir vaqtning o‘zida foydalaniladigan maydon hajmi va fayllar soni kvotasini o‘rnatish mumkin. Birinchi kiritilgan cheklovlar ishlaydi.

Fayl mazmunini o‘zgartirgandan keyin, yozish va redaktordan chiqish yetarli.

Joriy kvota haqida ma’lumotlarni olish uchun quota dasturidan foydalanish mumkin. Joriy holatni ko‘rish uchun ixtiyoriy foydalanuvchi uni chaqirishi mumkin.

Guruh kvotasini ko‘rish uchun -g parametridan foydalanish kerak:

```

# quota -g students
Disk quotas for group students (gid 1000):
Filesystem blocksquota limit grace
files quota limit grace
/dev/hda7 10022 0 100000 594 0 0
#

```



1. Bitta katologdan foydalanishga cheklovlarni kiritish mumkinmi?
2. Yangi versiyalarga kvota uchun qaysi modulni qo‘llash

kerak?

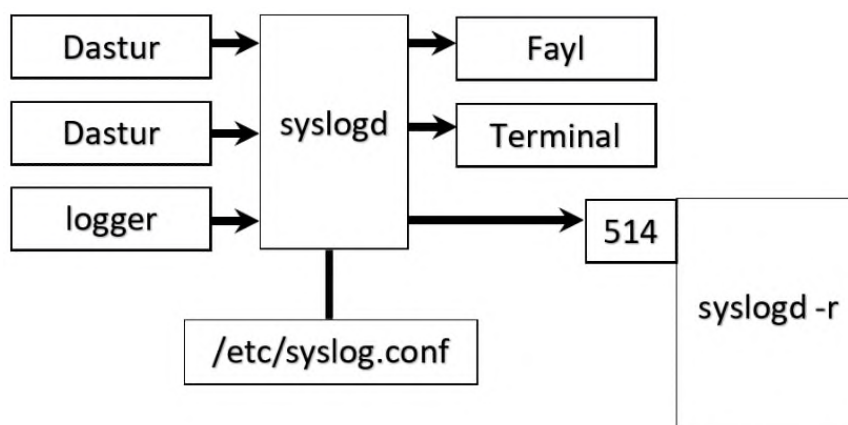
3. Qaysi dastur kvota haqidagi ma'lumotlarni tiklaydi?
4. Foydalanuvchi va guruh kvotalarini o'zgartirish uchun qaysi dastur qo'llaniladi?
5. Kvotaning joriy holatini tekshirish uchun qaysi dastur (dasturlar)dan foydalaniladi?

## 4-BOB. LOG FAYLLARI BILAN ISHLASH

### 4.1-§. Syslog tizimi

Syslog tizimi tarkibiga quyidagilar kiradi:

- syslogd dasturi.
- syslogd dasturidan xabarlarni jo‘natishda qo‘llaniladigan dastur kutubxonasi.
- syslogd dasturiga xabarlarni yuborish uchun qo‘llaniladigan logger dasturi.



Dasturdan olingan ma’lumot syslogd dasturida, uning konfiguratsiya fayli /etc/syslog.conf-da yozilgan filtrga mos ravishda filtrlanadi. Filtrlangan xabarlar quyidagi larga uzatiladi:

- faylga.
- terminal ekraniga.
- tarmoqdan syslogd dasturi ishlayotgan boshqa kompyuterga.



*Barcha dasturlar syslog tizimi xizmatlaridan foydalanmasligini bilishimiz kerak. Masalan, Apache WEB serveri, Squid RroXu serveri, Samba dasturlari foydalanmaydi.*

### 4.2-§. Syslogd dasturining konfiguratsiya fayli

Syslogd dasturining konfiguratsiya fayli - /etc/syslog.conf-da taqdim etilgan ma’lumotlar oqimidan zarurlarini ajratib olish uchun filtr tavsiflanadi va ajratilgan ma’lumotlar qaerga joylashishi ko‘ratiladi.

syslog.conf faylining satrlari formati quyidagicha:

**filtr «tabulyatsiya» harakat**

Bu yerda - «filtr» - qanday xabarlar tanlanishi bo'yicha filtr, «harakat» - filtr orqali tanlab olingan ma'lumotlar bilan qanday amallar bajarilishi.



*Bu ikkita maydon albatta ajratilib ko'rsatilishi uchun bitta yoki undan ko'p tabulyatsiya belgisi qo'yilishi kerak. Agar hech bo'lmaganda bitta probel uchrasa, bu xato hisoblanadi.*

# - izoh qo'yish belgisi.

**«Harakat» maydoni.** «harakat» maydonida quyidagilardan foydalanish mumkin:

- faylga yo'l.
- nomlangan kanallar (named pipes)
- terminallar va konsollar
- masofadagi kompyuterlar
- foydalanuvchilar ro'yxati
- "\*" belgisi.

**Faylga yo'l.** Agar «harakat» maydonida olingan ma'lumotlarni log faylining oxiridan qo'shish zarurligini ko'rsatish kerak bo'lsa, ushbu faylga to'liq yo'l ko'rsatilishi shart:

/var/log/file

Ba'zi syslog versiyalarida bu yozuvni quyidagicha ham ko'rsatish mumkin:

-/var/log/file

Odatda syslogd dasturiga beriladigan ma'lumotlar darhol, buferga olinmasdan (sinxron ravishda) qattiq diskdagi faylga saqlanadi. Bu xususiyat tizimning ishdan chiqish holatlarida ma'lumotlarni log faylida saqlashga ba'zi kafolatlarni beradi. Lekin boshqa bir muammo paydo bo'lishi mumkin – katta hajmdagi ma'lumotlar kelib tushganda syslog tizimi server ishini to'xtatib qo'yadi, ya'ni buferlamasdan diskka yozish amali bajarilishi uchun ko'p vaqt talab qiladi.

Zamonaviy syslog tizimi versiyalarida fayl nomidan oldin “-” belgisidan foydalanish mumkin. Bu belgi syslogd dasturiga olingan ma'lumotlarni darhol diskka yozmaslikni va fayl tizimining standart buferlashini qo'llash mumkinligini bildiradi. Albatta, tizimning ishdan chiqishi natijasida ma'lumotlarning bir qismi yo'qolishi mumkin, lekin buning uchun server ishida to'xtalish bo'lmaydi.

**Nomlangan kanallar.** Bunday imkoniyatlar juda ham kam holatlarda qo'llaniladi va asosan otladka rejimi uchun mo'ljallangan.

Dastlab mkfifo dasturi yordamida nomlangan kanal faylini hosil qilib olish kerak. Shundan so'ng konfiguratsiya fayli /etc/syslog.conf dagi "harakat" maydonida quyidagi satrdan foydalanish mumkin bo'ladi:

```
|/yo'l/fayl/ga
```

**Terminallar va konsollar.** Agar olingan ma'lumotlar terminal ekraniga chiqarilsa, "harakat" maydonida tanlab olingan terminal yoki konsolga mos keluvchi qurilma faylini ko'rsatish kerak bo'ladi.

```
/dev/tty9
```

**Masofadagi kompyuter.** Agar "harakat" maydonida quyidagi satrlardan biri yozilsa:

```
@195.158.1.1
```

```
yoki
```

```
@log.samtuit.uz
```

U holda filtrda tanlab olingan ma'lumotlar «@» belgisidan keyin ko'rsatilgan kompyuterga tarmoqdan uzatiladi. Masofadagi kompyuterda syslogd dasturi ishlab turgan bo'lishi shart.



*Masofadagi kompyuterdagi Syslogd dasturi –r parametri bilan ishga tushirilgan bo'lishi kerak, chunki u tarmoqdan xabarlarini 514 porti bilan qabul qilishi mumkin. Ma'lumotlarni yuborish uchun UDP protokoli qo'llaniladi.*

**Foydalanuvchilar ro'yxati.** Odatda tadqiqiy (kritik) xabarlar root foydalanuvchisi ishlayotgan terminalga chiqadi. Agarda filtrda tanlab olingan xabarlar aniq bir foydalanuvchining terminal ekraniga tushishini istasak, u holda "harakat" maydonida vergul bilan ajratilgan ushbu foydalanuvchilar ro'yxatini keltirishimiz kerak bo'ladi.

«\*» **belgisi.** Agar "harakat" maydonida «\*» belgisi yozilsa, mos filtrlar orqali tanlangan xabarlar, ayni vaqtda tizimda ishlayotgan barcha foydalanuvchilarning terminallariga chiqariladi.

#### **4.3-§. Xabarlarining muhimlik darajasi va vositalari.**

**Vositalar:** auth, authpriv, cron, daemon, kern, lpr, mail, news, syslog, user, uucp, local0 - local7

**Muhimlik darajasi:** debug, info, notice, warning, err, crit, alert,

emerg

Syslog tizimiga qanday ma'lumotlar berilishini dasturni yozayotgan dasturchi hal qiladi. Tizim administratorining vazifasi – barcha ma'lumotlar oqimidan, unga qiziqarli bo'lgan xabarlarni tanlashi va kerakli faylga joylashtirishi kerak.

Agar foydalanuvchi syslog tizimiga xabarlarni yuborish funksiyasidan foydalanayotgan bo'lsa, u xabar matnididan tashqari quyidagi ikkita parametrni ko'rsatishi shart:

- vositalar (facility).
- xabarning muhimlik darajasi (priority).

Bunday parametrlar sifatida oldindan aniqlangan so'zlardan foydalanish mumkin.

Quyida xabarlarni filtrlash va jo'natishda foydalaniladigan vositalar ro'yxati keltirilgan:

<b>Vositalar</b>	<b>Tavsifi</b>
auth	Xabar autentifikatsiya bilan bog'liq, tarkibida maxfiy ma'lumot yo'q
authpriv	Xabar autentifikatsiya bilan bog'liq, tarkibida maxfiy ma'lumot bor
cron	cron, anacron, at va boshq. dasturlarda jadval bo'yicha bajariladigan dastur xabari
daemon	Turli dasturlardagi xabarlar
kern	Yadro xabari
lpr	Chop etish tizimining xabari
mail	Pochta tizimining xabari
mark	Syslog tizimining ichki vositasi bo'lib, vaqtincha metka (nishon)larni generatsiyalashga mo'ljallangan. Buni dasturlarda ishlatmaslik kerak.
news	Yangiliklar serverining xabari
syslog	Syslog tizimining ichki xabari
user	Foydalanuvchi dasturlari tomonidan hosil qilingan ixtiyoriy xabar
uucp	Uucp tizimining xabari
local0 ... local7	Standart vositalar turiga kirmaydigan dasturlar ishlatishi mumkin bo'lgan qo'shimcha vositalar

Quyidagi jadvalda filtrlarni tavsiflashda va syslog tizimiga xabar jo'natuvchi dasturlarda qo'llash mumkin bo'lgan muhimlik darajalari

keltirilgan.

Daraja	Tavsifi
debug	Otladka ma'lumotiga yordamchi xabarlar.
into	Axborot xabarlari.
notice	E'tibor qaratish uchun oddiy hodisalar
warning	Ogohlantiruvchi xabar.
err	Me'yorda ishlashga to'siq bo'ladigan xatoliklar
crit	err ga nisbatan jiddiyroq xatolik
alert	Tezkor javob talab qiladigan hodisa
emerg	Tizimni falajlovchi xatoliklar
none	Darajasi yo'q. Filtrlarda istisnolarni o'rnatish uchun ishlatiladi.

Bulardan tashqari syslog tizimining zamonaviy versiyalarida oldingi (eski) versiyalari bilan muqobillikni ta'minlash uchun quyidagi muhimlik darajalari kiritilgan:

- warn – warning-ning aynan o'zi.
- error – err-ning aynan o'zi.
- panic – emerg-ning aynan o'zi.

«fil'tr» **maydoni.** «fil'tr» maydonida axborot oqimidan faqat zarur bo'lgan aniq qismlarini ajratib olish uchun filtrlar yoziladi.

Filtrda axborotlarni tanlab olish uchun shart sifatida faqat vositalar va muhimlik darajalaridan foydalanish mumkin.



*Syslogd dasturiga kelib tushgan ma'lumotlarda konfiguratsiya faylida tavsiflangan barcha filtrlar tekshiriladi. Ya'ni, aynan bitta xabar turli log fayllariga tushib qolishi mumkin.*

### Filtrlarni yozishga misollar:

Filtr	tavsifi
mail.debug	Muhimlik darajasi debug va undan yuqori bo'lgan mail vositasining xabari
mail.!emerg	Muhimlik darajasi emerg dan kichik bo'lgan mail vositasining xabari.
mail.news.=notice	Muhimlik darajasi faqat notice bo'lgan mail yoki news vositasining xabari.
*.notice;mail.none	mail vositasi istisnosi bilan muhimlik darajasi notice va undan yuqori bo'lgan barcha



Bitta satrda «;» belgisi bilan ajratilgan bir nechta filtrlarni ko'rsatish mumkin.

Quyida /etc/syslog.conf faylidan olingan real misolni qarab chiqamiz:

```
*.info;*.!warn;\  
authpriv.none; cron.none; mail.none;  
news.none -/var/log/messages
```

Barcha vositalarning xabarlari muhimlik darajasi info (\*.info) bilan boshlanib, warn muhimlik darajasi bilan oxirgisini hisobga olmasdan tugallanadi. Istisno tariqasida authpriv, cron, mail va news vositalarining xabarlari /var/log/messages faylining oxiridan qo'shiladi.

```
*.warn;\  
authpriv.none; cron.none; mail.none;  
news.none -/var/log/syslog
```

Barcha vositalarning muhimlik darajasi warn bilan boshlanadi. Istisno tariqasida authpriv, cron, mail va news vositalarining xabarlari /var/log/syslog faylining oxiridan qo'shiladi.

```
*.=debug -/var/log/debug
```

Barcha vositalarning otladka ma'lumotlari /var/log/debug faylining oxiridan qo'shiladi.

```
authpriv.* - /var/log/secure
```

Ixtiyoriy muhimlik darajasida authpriv vositasining xabarlari /var/log/secure faylining oxiridan qo'shiladi.

```
cron.* - /var/log/cron
```

Jadval (reja) bo'yicha topshiriqlarni bajarish tizimining xabarlari /var/log/cron faylining oxiridan qo'shiladi.

```
mail.* - /var/log/maillog
```

Pochta tizimining barcha xabarlari /var/log/maillog fayliga joylashtiriladi.

**\*.emerg \***

cron, anacron va at tizimlarining barcha xabarlari /var/log/cron fayliga joylashadi.

**\*.emerg \***

Kritik xabarlar joriy vaqtda tizimda ishlayotgan barcha foydalanuvchilar terminaliga chiqariladi.

**uucp,news.crit -/var/log/spooler**

Muhimlik darajasi crit va undan yuqori bo'lgan uucp va news vositalaridan yuborilgan xabarlar /var/log/spooler faylining oxiridan joylashtiriladi.

#### **4.4-§. Syslog tizimi ishining ba'zi xususiyatlari**

syslog tizimi bilan ishlashda uning ba'zi xususiyatlarini hisobga olish kerak. Birinchi navbatda, bu syslogd dasturi log fayllari bilan qanday ishlashiga bog'liq.

Agar faylning tarkibi tahrirlangan yoki o'chirilgan va uni yangidan hosil qilingan bo'lsa, syslogd dasturi ular bilan ishlamaydi va ma'lumotlar yo'qotiladi. Agar fayllarga o'zgartirishlar kiritilgan yoki fayllar almashtirilgan bo'lsa, syslogd dasturiga HUP signali yuborilishi kerak:

**killall -HUP syslogd**

Agar konfiguratsiya fayli /etc/syslogd.conf ning mazmuni o'zgartirilgan bo'lsa, HUP signali syslogd dasturiga o'zining konfiguratsiya faylini o'qishi uchun yuborilishi kerak.

Konfiguratsiya faylida aniqlanadigan barcha fayllarni syslogd ning o'zi hosil qiladi.

Agar log faylining tarkibini tozalash (o'chirish) zarur bo'lsa, uni quyidagi komanda satri orqali bajarish mumkin:

**echo -n > file**

Bunday holda faylning mazmuni o'zgartirilgan bo'lishiga qaramasdan, dasturga HUP signalini yuborish kerak emas, dastur ushbu fayl bilan ishlashni davom ettiradi.

Syslogd dasturi bilan ishlashning yana bir xususiyati – bu IP adreslarni kompyuter nomiga akslantirish hisoblanadi. Dastur bunday

maqsad uchun DNS ni ishlatmaydi. Agar log fayllarda IP adreslarning o'rniga kompyuter nomlarini ko'rsatish kerak bo'lsa, buni /etc/hosts faylida tavsiflashimiz kerak bo'ladi.

1. syslog tizimida filtrdan tanlab olingan ma'lumotlar qaerga tushishini (joylashishini) sanab bering?

2. syslog tizimida Linux yadrosiga uzatiladigan ma'lumotlarni ajratib olish uchun qanday vositalarni qo'llash kerak?



3. mail.info; mail.!err fil'tri nimani anglatadi?

4. Log fayllarini tahrirlash nima uchun tavsiya etilmaydi?

5. Log faylining mazmunini syslogd dasturiga qo'shimcha HUP signalini yubormasdan qanday tozalash mumkin?

## 4.5-§. Tizim foydalanuvchilari hisobi

Linux administratori tizimda qachon va qanaqa foydalanuvchilar ishlaganini doimiy tekshirib turishi shart.

Foydalanuvchilar ishini nazorat qilishining ikkita ko‘rinishi mavjud:

- Joriy vaqtda qanday foydalanuvchilar tizimda ishlayapti.
- Qachon va qanday foydalanuvchilar tizimda ishlagan.

### **Joriy vaqtda qanday foydalanuvchilar tizimda ishlayapti.**

Joriy vaqtda tizimda ishlayotgan foydalanuvchilar hisobini yuritish uchun administrator quyidagi dasturlardan foydalanishi mumkin:

- **who**
- **w**
- **ps**

who va w dasturlari faqat tizimda ishlab turgan foydalanuvchilarni ko‘rsatadi. Ma’lumotning to‘liqlik darajasi qo‘llanilayotgan dasturga bog‘liq.

```
#who
ilhom pts/0 Mar 5 13:59
ilhom pts/1 Mar 5 14:03
#w
18:33:43 up 4:36, 2 users, load average:
0.05,0.05,0.01
USER TTY FROM LOGIN@ IOLE JCPU WHAT
ilhom pts/0 -1:59pm 4:36m 0.00c 0.23s
kwwrited
ilhom pts/1 -2:03pm 0.00s 0.07s 0.04s
/bin/bash
```

Ushbu dasturlar taqdim etgan barcha ma’lumotlar /var/run/utmp binar faylida saqlanadi.

Tizimdagi foydalanuvchi qanday dasturdan foydalanayotganini aniqlash uchun ps dasturidan foydalanish mumkin:

```
ps -u user
```

**Tizimda oldin ishlagan foydalanuvchilar hisobi.** Tizimda kim, qachon ishlaganini ko‘rish uchun quyidagi dasturlar qo‘llaniladi:

- last
- lastlog

Bu ikkala dastur ham /var/log/wtmp binar faylida saqlanayotgan ma'lumotlarni tasvirlab beradi.

last dasturidan foydalanishda parametr sifatida foydalanuvchi loginini ko'rsatish mumkin, bunday holda aynan ushbu foydalanuvchi haqidagi ma'lumotlarni olish mumkin bo'ladi

```
# last ilhom
ilhom pts/1 Fri Mar 5 14:03 still logged in
ilhom pts/0 Fri Mar 5 13:59 still logged in
ilhom pts/1 Fri Mar 5 10:42 - 11:02 (00:20)
ilhom pts/0 Fri Mar 5 10:42 - 11:52 (01:09)
ilhom pts/1 Wed Mar 3 18:07 - 21:07 (03:00)
ilhom pts/0 Wed Mar 3 18:07 - 21:07 (03:00)
ilhom pts/2 Wed Mar 3 17:06 - crash (00:59)
ilhom pts/1 Wed Mar 3 16:30 - crash (01:35)
ilhom pts/1 Wed Mar 3 15:55 - 16:01 (00:06)
ilhom pts/0 Wed Mar 3 15:55 - crash (02:10)
ilhom tty2 Tue Mar 2 18:59 - 18:59 (00:00)
```

Misoldan ko'rinib turibdiki, bu faylda foydalanuvchi qachon tizimga kirgan va qachon tizimdan chiqib ketganligi haqidagi ma'lumotlar saqlanadi. Shuningdek, foydalanuvchi tizimdan qachon me'yor bo'yicha chiqmaganligi ham ko'rsatiladi. Bunday holat crash so'zi yordamida ko'rsatilgan satrlarda taqdim etilgan. Shu bilan birgalikda foydalanuvchi qancha vaqt tizimda ishlagani ham ko'rsatib o'tiladi.

Foydalanuvchi ishlaridan tashqari wtmp faylida tizimning qayta yuklanganligi haqidagi ma'lumotlar ham saqlanadi. Foydalanuvchi logini o'rniga reboot so'zi ko'rsatiladi.

#### 4.6-§. Xavfsizlik masalalari

Afsuski, administrator log fayllari va wtmp fayliga to'liq tayanolmaydi. Agar buzg'unchilar tizimga kirish ruxsatini olsa va uning vazifasi faqat ma'lumot to'plash bo'lsa, u ushbu fayllarning mazmunini o'zgartirishi mumkin.

Tizimga tashqaridan "hujum-bosim" bo'layotganligini aniqlashga yordam beradigan turli xil usullar mavjud.

Agar bir nechta Linux serverlar mavjud bo'lsa, ulardan birini tarmoqda log fayli uchun markaziy server sifatida ajratish kerak bo'ladi. Ushbu serverga (odatda ichki tarmoqda joylashgan) hujumlar ehtimoli yetarlicha kichik bo'ladi. Bundan tashqari bu server nafaqat Linux, balki ixtiyoriy UNIX kompyuterdan, shuningdek, syslog tizimini qo'llab-quvvatlovchi ixtiyoriy qurilmadan axborotlarni qabul qilishi mumkin, masalan, Cisco firmasi.

Agar markaziy serverni ajratish imkoni bo'lmasa, syslogd dasturiga yondoshuvning quyidagi xususiyatlarini hisobga olish kerak:

- agar log fayl tahrirlansa, dastur bu fayl bilan ishlashni to'xtatadi.

- agar buzg'unchilar ushbu faylning mazmunini o'zgartirsa, HUP signalini yuborishi yoki dasturni qayta yuklashi kerak bo'ladi.

Tizimni yuklash, qayta yuklash va to'xtatishning barcha holatlarini Syslogd dasturi log faylida tasvirlab boradi. Va shunga o'xshash xabarni /var/log/messages faylida ham ko'rishimiz mumkin:

```
Mar 6 16:29:02 master syslogd 1.4.1:
restart
yoki
Mar 6 16:30:37 master syslog: start syslogd
succeeded
```

Bunday xabarlardan kelib chiqadiki, katta ehtimol bilan faylning mazmunini kimdir o'zgartirgan.

Bundan tashqari, B laboratoriya ishida yaratilgan /var/log/auth.notice faylining ham mazmunini nazorat qilish mumkin. Ya'ni, ushbu faylga tushgan oxirgi ma'lumotlar asosida, oxirgi marta kim tizimga kirganligi haqida axborotlarni olish mumkin bo'ladi.

Superfoydalanuvchi tizimiga kirishlarni nazorat qilish uchun auth.notice faylidan quyidagi mazmundagi satrni qidirishda qo'llaniladigan skript yozish kerak bo'ladi:

```
Mar 5 18:00:49 master - root[5074]: ROOT
LOGIN ON tty2
```

Agar bunday mazmundagi satr topilsa, u holda skript tashqi pochta serverida bo'lgan elektron pochtaga ushbu faylning mazmunini yuboradi. Faylning o'zi esa tozalanadi.

Log faylidagi xabarni avtomatik tekshirish g'oyasining boshqa

bir tatbiqi – bu logwatch<sup>6</sup> dasturi hisoblanadi. Bu dastur jimlik holatida aynan ushbu serverning root foydalanuvchisi elektron pochta-siga hisoblarni yuboradi. Boshqa elektron pochta-ni ko‘rsatish uchun esa, /var/log.d/conf/logwatch.conf faylidagi “MailTo” parametrini o‘zgartirish kerak.

Yuqorida ko‘rsatilgan skript va logwatch dasturi demon hisoblanmaydi, ularni davriy aniqlik bilan ishga tushirib turish zarur.



1. Qaysi dasturlar yordamida qanday foydalanuvchi va qaerdan ulangan holda tizimga kirganligini ko‘rish mumkin?

2. Qaysi dasturlar yordamida joriy vaqtda foydalanuvchi qanday dasturda ishlayotganligini ko‘rish mumkin?

3. Qanday yo‘l bilan oxirgi foydalanuvchilarning tizimga kirgan vaqtini aniqlash mumkin?

4. Qanday yo‘l bilan tizimning oxirgi qayta yuklangan vaqtini aniqlash mumkin?

---

<sup>6</sup> Bu Slackware Linux-da qo‘yilmagan

## 4.7-§. Dasturlarning jadval bo'yicha bajarilishi

Oldingi laboratoriya ishida /var/log/auth.notice faylining mazmunini tekshiruvchi skript yozilgan edi. Bu skriptni o'rtacha to'rt minutda bir marta davriy bajarilishi yo'lga qo'yish, ya'ni dasturning bajarilishini rejalashtirish kerak bo'ladi.

Linux-da dasturlarning oldindan aniqlangan jadval asosida bajarilishini tashkillashtirish uchun uchta dasturdan foydalaniladi:

- cron (Vixie yoki Dilans)
- Anacron
- at

**Cron.** Bu dastur topshiriqlarning jadval asosida bajarilishini tashkillashtirish uchun qo'llaniladi. Quyida *CRON* tizimining klassik variantini ko'rib chiqamiz.

crond dasturi /var/spool/cron katologidan crontab faylini izlaydi.

crond dasturining barcha konfiguratsiya fayllari o'zining sintaksisiga juda o'xshaydi.

min hour dayOfMonth Month dayOfWeek process

Bu yerda birinchi beshta topshiriqning bajarilishi vaqtini aniqlovchi maydonlarda quyidagi raqamlardan foydalanish mumkin:

- min - 0-59
- hour - 0-23
- dayOfMonth -1-31
- Month – 1-12
- dayOfWeek - 0-7 (0 va 7 lar yakshanba kunlari)

Topshiriqlarning bajarilishi vaqtini yozishga misollar:

**30 1 \* \* \* prog**

prog dasturi oyning va haftaning ixtiyoriy kuni 1:30 da bajariladi. "\*" belgisi ixtiyoriy qiymat ekanligini bildiradi.

**30 1,2,3,4,5,6 \* \* \* prog**

Bu misolda dastur 1:30, 2:30, 3:30, 4:30, 5:30 va 6:30 larda qayta bajariladi. Ikkinchi maydonda vergul orqali ajratilgan qiymatlar keltirilgan.

**30 1-6 \* \* \* prog**

Bu misolda ham dasturning bajarilishi vaqti oldingi misoldagigi o'xshash bo'ladi. Lekin bunda qiymatlar interval ko'rinishda berilganligi bilan farq qiladi.

Interval va sanoqli qiymatlar ko'rinishlaridan aralash shaklda foydalanish mumkin:



**30 1-6,18,21 \* \* \* prog**

Xuddi shunday shaklda topshiriqlarning davriy bajarilishi ko'rsatish mumkin. Masalan, dastur har charok minutda bajarilishi kerak bo'lsin. U holda jadval faylidagi yozuvni quyidagicha kiritish kerak:

**\*/4 \* \* \* \* prog**

Vaqtning tavsiflovchi barcha maydonlar mantiqiy VA dan foydalangan holda birlashtiriladi. Faqat istisno tariqasida "dayOfMonth" va "dayOfWeek" maydonlari mantiqiy YOKI yordamida birlashtiriladi.

Aytmalik faylda vaqt quyidagi ko'rinishda berilgan bo'lsin:

**0 12 12,14 \* 1,6**

Bunday holda dastur oyning 12 va 14 sanalarida YOKI har dushanba va shanba kunlari bajariladi.

Bu xususiyat ikkala maydonni ham bitta satrda yozmaslik kerakligiga e'tibor qaratish zarurligini keltirib chiqaradi. Yaxshisi, yuklanadigan dasturlarning bajarilishini ikkita satrda yozish kerak.

Agar ishga tushirilgan dastur biror narsani yoki xatolarni chiqarishga harakat qilsa, bu ma'lumotlar crontab faylining egasi bo'lgan foydalanuvchiga xat orqali yuboriladi.

Agar tizimning oddiy foydalanuvchisi o'zining topshiriqlar jadvalini tashkil etishni xohlasa, u crontab dasturidan foydalanishi shart. Bu dastur /var/spool/cron katalogida foydalanuvchi nomi bilan jadval faylini hosil qiladi.

**crontab [ -u user ] file**

**crontab [ -u user ] { -l | -d | -e }**

Crontab dasturining parametrlari:

<b>Parametrlar</b>	<b>Tavsifi</b>
-u user	Dasturlari ushbu jadval fayli bo'yicha ishlaydigan foydalanuvchini aniqlaydi. Jimlik holati bo'yicha dastur joriy foydalanuvchi fayllari bilan ishlaydi. Faqat root foydalanuvchisi uchun kirishga ruxsat beriladi.
-l	Foydalanuvchining topshiriqlar jadvalini ko'rsatish
-e	Topshiriqlar fayli mazmunini tahrirlash
-d	Foydalanuvchi topshiriqlari faylini o'chirish
file	Joriy topshiriqlar faylini ko'rsatilgan faylga

Dasturni “-e” parametri bilan yuklashda faylni tahrirlash uchun vi redaktori chaqiriladi. Agar boshqa redaktordan foydalanish zarurati bo‘lsa, u holda VISUAL yoki EDITOR muhitining o‘zgaruvchisini aniqlab olish shart. Masalan, mcredit redaktoridan foydalanmoqchi bo‘lsak, zarur o‘zgaruvchi quyidagi ko‘rinishda aniqlanadi:

```
export VISUAL=/usr/bin/mcredit
```



*/var/spool/crontab* katologida joylashgan bajrilish jadvali fayllarining mazmunini matnli redaktorlar yordamida o‘zgartirish taqiqlanadi. Agar shunday qilinsa, crond dasturi bu fayllarni ishlatmaydi. Bunday fayllar mazmunini o‘zgartirish uchun albatta crontab dasturidan foydalanish kerak

Slaskware Linuxda root foydalanuvchi uchun oldindan aniqlangan jadval fayli mavjud. Uning mazmuni quyidagi ko‘rsatilgan. Izohlar va bo‘sh satrlar o‘chib tashlangan.

```
47 * * * * /usr/bin/run-parts /etc/cron.hourly
1> /dev/null
40 4 * * * /usr/bin/run-parts /etc/cron.daily
1> /dev/null
30 4 * * 0 /usr/bin/run-parts /etc/cron.weekly
1> /dev/null
20 4 1 * * /usr/bin/run-parts /etc/cron.monthly
1> /dev/null
```

Misoldan ko‘rinib turibdiki, run-parts dasturi soatiga bir marta, har kuni, har hafta va har oyda ishga tushadi.

**Anacron.** Yuqorida tavsiflangan cron dasturi UNIX ning barcha versiyalariga qo‘yilgan standart dastur hisoblanadi. Bu dastur sutkasiga 24 soat ishlaydigan serverlarda foydalanish uchun mo‘ljallangan. cron dasturida muhim bir kamchilik bor. Agar kompyuter o‘chirilsa, va aynan shu vaqt oralig‘ida qaysidir dasturlarning ishga tushishi rejalashtirilgan bo‘lsa, bu dasturlar yuklanmaydi.

Ishchi stantsiyalarida boshqa – anacron dasturidan foydalanish tavsiya etiladi. Bu dastur cron dan farqli ravishda hatto kompyuter o‘chirilgan vaqt oralig‘ida dasturlarni yuklash rejalashtirilgan bo‘lsa,

ushbu dasturlarni ishga tushirishi mumkin. Ya'ni, kompyuter qo'shilgandan keyin anacron dasturi topshiriqlar ro'yxatini qarab chiqadi va agar bajarilmay qolgan topshiriqlar topilsa, ularni ishga tushiradi.

Anacron dasturining konfiguratsiya fayli /etc/anacrontab bajarilishi zarur bo'lgan topshiriqlar jadalini o'zida saqlaydi.

Anacrontab faylining formati:

**davr kechikish topshiriq-identifikatori  
komanda**

“Davr” maydoni kunlar sonini aniqlovchi sonni saqlaydi. Aynan shunday namunada topshiriqni bajarish davriyligi ko'rsatiladi. Masalan, agar bu maydonda 7 soni yozilgan bo'lsa, u holda dastur yetti kunda bir marta bajariladi.

“Kechikish” maydonida minutlar sonini aniqlovchi son yoziladi. Bu maydon yordamida dastur qanday kechikish bilan bajarilishi ko'rsatiladi. Masalan, birinchi maydonda 1 soni, ikkinchi maydonda 15 soni yozilgan bo'lsa, bu anacron dasturi yuklangandan keyin oxirgi marta ko'rsatilgan dastur qachon ishlaganligini tekshirishni bildiradi. Agar dastur bir kundan ortiq vaqt oldin bajarilgan bo'lsa, anacron dasturi 15 minut kechikish bilan ko'rsatilgan dasturni ishga tushiradi.

“Topshiriq-identifikator” maydonida anacron dasturi xabarni chiqarish uchun fayl nomi sifatida foydalanadigan unikal so'z bo'lishi kerak.

“Komanda” maydonida anacron dasturi ishga tushirishi zarur bo'lgan dastur nomi yoziladi.

anacron dasturi o'rnatilgandan keyin uning konfiguratsiya faylining mazmuni quyidagicha bo'ladi:

```
SHELL=/bin/sh  
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/  
bin:/usr/sbin:/usr/bin  
1 65 cron.daily run-parts /etc/cron.daily  
7 70 cron.weekly run-parts /etc/cron.weekly  
30 75 cron.monthly run-parts  
/etc/cron.monthly
```

Bu faylda ishga tushiriladigan dasturga beriladigan o'zgaruvchilar muhiti aniqlanadi. Shuningdek uchta satrda bajarilishi vaqti kuniga bir marta, haftasiga bir marta, 30 kunda bir marta kabi aniqlangan.

Anacron dasturining kamchiligi sifatida topshiriqning aniq bajarilish vaqtini ko'rsatish imkoniyati yo'qligini, shuningdek, takrorlanish davrining minimal qiymati bir kun ekanligini ko'rsatish mumkin.

Aynan cron dasturidagiga o'xshash bajarilayotgan topshiriq agar biror xabarni chiqarishi yoki xatolikni chiqarishga harakat qilsa, bu ma'lumotlar dasturni ishga tushirgan foydalanuvchining (odatda root) pochta-siga yuboriladi.

Anacron dasturi tizim yuklanishida ishga tushadi. Yuklanuvchi skriptga /etc/rc.d/init.d/anacron-ning ishga tushishida "-s" parametri beriladi. Bu anacron dasturi barchasini birdaniga ishga tushirmasdan ketma-ket bajarishini ta'minlaydi. Ya'ni, u oldin ishga tushirilgan topshiriq tugallanishini kutib turadi va keyingi topshiriqni ishga tushiradi.

Anacron maxsus fayldagi vaqtinchalik belgi (metka) bo'yicha topshiriqning oxirgi bajarilgan vaqtini aniqlaydi. Bu fayl /var/spool/anacron katalogida joylashgan bo'ladi.

**/etc/cron.\* katalogi va run-parts dasturi.** Crontab fayllari va /etc/anacrontab konfiguratsiya fayllaridan ko'rinib turibdiki, cron va anacron run-parts dasturini bajarish uchun yuklaydi. Dasturning parametri sifatida quyidagi kataloglardan biri keltirilgan edi:

- **cron.hourly**
- **cron.daily**
- **cron.weekly**
- **cron.monthly**

Run-parts dasturining vazifasi ko'rsatilgan kataloglardagi barcha bajariluvchi fayllarni ishga tushirishdan iborat bo'ladi.

Agar dasturlarning ishga tushirilish aniq vaqti ahamiyatga ega bo'lmasa, faqat dastur soatiga, kuniga, haftasiga yoki oyiga bir marta ishga tushishi muhim bo'lsa, u holda ko'rsatilgan kataloglardan bittasiga dasturlarni ishga tushiruvchi skriptni joylashtirib qo'yish yetarli bo'ladi.



*Agar faqat anacron dasturidan foydalanilsa, u holda bu dasturda ishga tushirishning takrorlanish davri uchun eng kichik kvant davr bir kun ekanligini e'tiborga olish kerak bo'ladi. Shuning uchun ham cron-hourly katalogidan foydalanish shart bo'lmaydi.*

#### **4.8-§. at va atd dasturlari**

Cron tizimi va anacron dasturlari topshiriqlar (jarayonlar)ning aniq belgilangan vaqtlarda davriy bajarilishini ta'minlab beradi. atd dasturi esa dasturlarning aniq belgilangan vaqtda bir marta bajarilishini ta'minlash uchun mo'ljallangan.

Topshiriqlarning bajarilishini navbatga qo'yish uchun foydalanuvchilar at dasturini qo'llashlari mumkin.

Bu dasturni ishga tushirishda topshiriqning bajarilishi shart bo'lgan vaqtni albatta ko'rsatish kerak bo'ladi.

**Topshiriqning bajarilishi vaqtini ko'rsatishga misollar.** At dasturida topshiriqning bajarilishi vaqtini ko'rsatishning bir nechta usullari mavjud:

- NN:MM formatidagi vaqt. Masalan, 18:00.

- NN:MM MM/DD/YY formatidagi vaqt. Masalan, 12:28 16/08/15.

- now kalit so'zini qo'llash orqali nisbiy vaqt. Masalan, now + 3 hours. Vaqt qiymatlari sifatida minutes, hours, days yoki weeks kabi kalit so'zlaridan ham foydalanish mumkin.

- Nisbiy vaqt. Masalan, 4pm + 3 days.

At dasturi ishga tushirilgandan keyin, foydalanuvchi tomonidan atd dasturini ishga tushiruvchi komandani kiritishi mumkin bo'lgan qo'shimcha taklif chiqariladi. Dastur bir necha marta bajarilishi (komandalar bir nechta kiritilishi) mumkin, komandalarni kiritishni tugallash uchun, satr boshida EOF belgisiga mos keluvchi Ctrl+D tugmalar kombinatsiyasini ishlatish kerak bo'ladi.

Oldindan atd dasturini ishga tushiradigan komandalar ro'yxatidan tashkil topgan fayl yaratish mumkin. Bunday holda at dasturini ishga tushirishda -f parametridan keyin ushbu faylning nomini ko'rsatish kerak bo'ladi.

#### **4.9-§. Navbatning bajarilishi vaqtida topshiriqlarni kuzatish**

Atd dasturi bajarilayotganda navbatda turgan topshiriqlarni ko'rish uchun atq dasturidan foydalanish mumkin.

Afsuski, bu dastur bajariladigan topshiriqlar ro'yxatini ko'rsatmaydi. U faqat topshiriqlar nomerini ko'rsatadi.



Agar topshiriq bajarilayotganda qanday dastur ishlayotganligini ko'rmqchi bo'lsak, u holda /var/spool/atjobs katologida joylashgan fayllar mazmunini ko'rish kerak bo'ladi.

**Navbatdan topshiriqni o'chirish.** Navbatdan topshiriqni o'chirish uchun atrm dasturidan foydalanish mumkin. Unga parametr sifatida atq dasturiga kiritilgan topshiriqning nomerini kirish kerak bo'ladi.

**Atd dasturining bajarilishiga cheklovlar.** Administrator at sistemasidan qaysi foydalanuvchi foydalanishi mumkin yoki mumkin emasligini ko'rsatish imkoniyati bor. Buning uchun ikkita fayl mavjud:

- /etc/at.allow
- /etc/at.deny

Qanday cheklov usuli qo'llanilganiga qarab faqat bitta fayldan foydalanish mumkin:

at.deny – ushbu faylda keltirilgan foydalanuvchilarga at sistemasidan foydalanish taqiqlanadi. Qolgan barcha foydalanuvchilarga mumkin.

at.allow – ushbu faylda keltirilgan foydalanuvchilarga at sistemasidan foydalanishga ruxsat berilgan. Qolgan barcha foydalanuvchilarga taqiqlanadi.

At ni o'rnatgandan keyin jimlik holati bo'yicha /etc/at.deny fayli bo'sh holda mavjud bo'ladi.



1. Cron tizimidan foydalanishda vaqtning qanday minimal oralig'idan foydalanish mumkin?
2. anacron dasturi yordamida dasturlarni ikki soatda bir marta bajariladigan qilish mumkinmi?
3. /etc/cron.monthly katologi nima uchun mo'ljallangan?
4. Topshiriqlarni atd dasturi yordamida ishlatishda qaysi dasturlar bajarilayotganligini qanday tekshirish mumkin?

#### 4.10-§. Log fayllarini almashtirish (rotatsiya)

Hamma dasturlar ham log faylining o'lchamini mustaqil ravishda cheklay olmaydi. Shuning uchun ham **Linux-ning** ixtiyoriy distributiviga fayllarni rotatsiyasi (o'rin almashtirish) uchun logrotate dasturi o'rnatilgan.

**Logrotate dasturi.** Fayllarning o'rnini almashtirish belgilangan shartlar bo'yicha amalga oshiriladi:

- kuniga bir marta (daily).
- haftasiga bir marta (weekly).
- oyiga bir marta (monthly).
- faylning belgilangan o'lchamidan oshib ketganda.

Logrotate dasturi quyidagi amallarni bajarishda qo'llaniladi:

- log fayllarining bir nechta nusxasini saqlashda.
- rotatsiyaga tushgan faylni pochta orqali jo'natishda. Agar fayllarning bir nechta nusxasi saqlanayotgan bo'lsa, u holda rotatsiya jarayonida o'chiriladigan fayl jo'natiladi.

- rotatsiyagacha yoki undan keyin turli dasturlarni ishga tushirishda.

**Dasturning konfiguratsiya fayli - /etc/logrotate.conf.** Bu faylda rotatsiyaga tushgan fayllarning jimlik holatidagi parametrlari tavsiflanadi.

Quyidagi jadvalda konfiguratsiya faylida qo'llash mumkin bo'lgan ba'zi parametrlar keltirilgan:

Parametr	Tavsifi
compress	Agar bu parametr ko'rsatilsa, rotatsiyadan keyin fayllar gzip yordamida siqiladi.
create [mode] [owner] [group]	Agar rotatsiya jarayonidan keyin ushbu parametrdan foydalansak, logrotate dasturi ko'rsatilgan parametrlar bilan fayl hosil qiladi.
include fayl yoki katolog	Ko'rsatilgan faylning mazmunini asosiy konfiguratsiya fayliga qo'shadi. Agar parametr sifatida katolog ko'rsatilsa, ko'rsatilgan katologdagi barcha fayllar konfiguratsiya fayliga qo'shiladi.
mail adres	Rotatsiyadan keyin fayl jo'natiladigan e-mail adresni aniqlaydi.
mailfirst	Log faylining birinchi nusxasi pochta orqali

	jo‘natiladi
maillast	Log faylining oxirgi nusxasi pochta orqali jo‘natiladi
missingok	Agar fayl mavjud bo‘lmasa, xatolik haqida xabar bermasdan keyingi faylni qayta ishlashga o‘tish
postrotate endscript	Rotatsiya jarayonidan keyin bajariladigan dasturlarni aniqlaydi.
prerotate endscript	Rotatsiya jarayonidan oldin bajariladigan dasturlarni aniqlaydi.
rotate son	Faylning nusxalari sonini aniqlaydi.
sharedscripts	Odatda postrotate yoki prerotate parametrlari yordamida aniqlanadigan barcha fayllar, har bir fayl uchun alohida bajariladi. Agar ushbu parametрни ko‘rsatsak, dasturlar barcha fayllar rotatsiyasi tugagandan keyin bir marta bajariladi.
weekly	Fayllar rotatsiyasi haftasiga bir marta amalga oshiriladi
size o‘lcham	Agar fayl ko‘rsatilgan o‘lchamdan oshib ketsa, rotatsiya amalga oshiriladi. O‘lchamni ko‘rsatishda “k” va “M” belgilaridan foydalanish mumkin.
daily	Fayl rotatsiyasi bir kunda bir marta bajariladi
monthly	Fayl rotatsiyasi bir oyda bir marta bajariladi

Tizim o‘rnatilgandan keyin olingan /etc/logrotate.conf faylining mazmuni quyida keltirilgan:

```

weekly
rotate 4
create
include /etc/logrotate.d
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}

```

Birinchi satrda fayllar rotatsiyasi haftada bir marta bajarilishi davri keltirilgan. Bu parametрни aniq faylning rotatsiyasi uchun qayta aniqlash mumkin.

Keyingi satrda jimlik holati bo‘yicha faylning 4 ta nusxasi



saqlanishi aniqlangan.

Keyin logrotate dasturi fayllarni mustaqil hosil qilishi ko'rsatilgan.

Include parametri /etc/logrotate.d katologida joylashgan barcha fayllarni biriktiradi.

Eng oxirida /var/log/wtmp faylining rotatsiyasi aniqlangan. Bu fayl har oyda bir marta rotatsiya qilinadi. Dastur uni 0664 kirish huquqi bilan hosil qiladi. U root va utmp guruhi foydalanuvchilariga tegishli bo'ladi. Faylning faqat bitta nusxasi saqlanadi.

Rotatsiya qilinadigan qolgan barcha fayllarning tavsifi /etc/logrotate.d katologidagi faylda joylashadi.

Masalan, /etc/logrotate.d/syslog faylining mazmunini qarab chiqamiz:

```
/var/log/cron /var/log/debug
/var/log/maillog /var/log/messages
/var/log/secure /var/log/syslog {
    sharedscripts
    postrotate
        /bin/kill -HUP cat /var/run/syslogd.pid
2>/dev/null 2>/dev/null || true
    endscript
}
```

Bu faylda syslog tizimi hosil qiladigan barcha fayllar keltirilgan. Rotatsiya jarayonidan keyin sysdogd dasturiga HUP signalini yuboradigan kill dasturi ishga tushiriladi. Jarayonning PID nomeri /var/run/syslogd.pid faylidan olinadi.

Shuningdek, sharedacripts parametri aniqlanishi, barcha fayllar rotatsiyasidan keyin kill dasturini faqat bir marta chaqirish amalga oshiriladi.

**Logrotate dasturini ishga tushirish.** Dastur demon ko'rinishda bajarilmaydi, ya'ni u har bir rotatsiya jarayoni amalga oshirilganda ishga tushirilishi kerak. Buning uchun cron yoki anacron tizimlaridan foydalanish mumkin.

Logrotate dasturi o'rnatilgandan keyin, /etc/cron.daily katologida rotatsiya dasturini ishga tushiradigan skript joylashadi. Ya'ni, dastur kuniga bir marta yuklanadi.



1. Log fayllarining rotatsiyasidan keyin qanday qilib

syslogd dasturini ular bilan ishlashga majburlash mumkin?

2. Rotatsiya jarayonidan keyin saqlanib qoladigan nusxalar sonini qanday aniqlash mumkin?

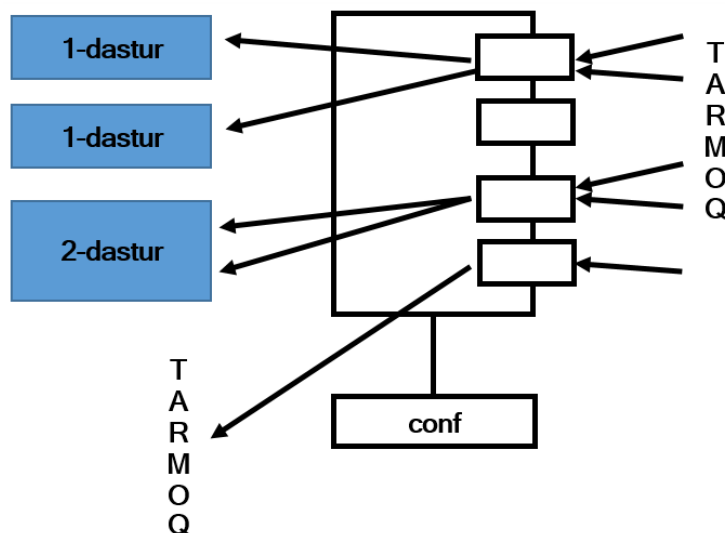
## 4.11-§. Tarmoq superserveri

Zamonaviy Linux distributivlarida tarmoq supervizori deb ataladigan ikki xil ko‘rinishdagi dasturlarni uchratish mumkin:

- **inetd**
- **xinetd**

Inetd dasturning klassik varianti hisoblanadi. Xinetd – bu UNIX tizimlarining tarmoq supervizorining keyingi rivojlanish kontsepsiyasi hisoblanadi.

Tarmoq supervizori ishga tushirilgandan keyin /etc/inetd.conf yoki /etc/xinetd.conf konfiguratsiya fayllarida tavsiflangan portlarni tekshirish uchun ochiladi. Agar portda bog‘lanishga so‘rov o‘tayotgan bo‘lsa, supervizor (konfiguratsiya faylida keltirilgan) zarur dasturni ishga tushiradi va unga bog‘lanish topshiriladi.



Bundan tashqari xinetd dasturi:

- TCP, UDP va RPC xizmatlariga kirishni nazorat qilishga ruxsat beradi.

- Xizmatlarga kirishni vaqt bo‘yicha cheklashi mumkin.

- logning kengaytirilgan imkoniyatlariga ega.

- “xizmatlarni rad qilish” turidagi hujumlardan himoya mexanizmiga ega.

- /etc/hosts.allow va /etc/hosts.deny fayllarini qo‘llashi mumkin, shuningdek, tcpd dasturini ishlatilishi mumkin.

- aniq tarmoq interfeysiga xizmatlarni bog‘laydi.

**Inetd dasturi.** Inetd dasturi – bu tarmoq supervizorining klassik ko‘rinishi.

Dasturning konfiguratsiya fayli - /etc/inetd.conf. Bitta yozuv

bitta satrga yoziladi. Bitta satrda vaziyatga bog‘liq holda oltita yoki yettita maydon bo‘lishi mumkin. Maydonlar probel yoki tabulyatsiya belgisi bilan ajratiladi.

- servis – servis nomi. Bu yerda faqat /etc/services faylida aniqlangan sevrslar yozilishi mumkin.

- bog‘lanish turi – mumkin bo‘lgan variantlar: stream, dgram va raw.

- protokol – bog‘lanishda qo‘llaniladigan transport protokolining nomi.

- bayroqlar (flag) – agar har bir bog‘lanishda yangi dasturni ishga tushirish kerak bo‘lsa, flag qiymati nowait-ga teng bo‘ladi. Agar bir nechta bog‘lanishlarni amalga oshirishda bitta dastur yuklangan bo‘lsa – uning qiymati wait bo‘ladi. Parametrga qo‘shimcha sifatida bir minutda maksimal bog‘lanishlar sonini aniqlovchi sonni yozish mumkin, masalan: nowait.30.

- ushbu jarayonni ishga tushirish huquqiga ega bo‘lgan foydalanuvchi. Guruhni ko‘rsatish ham mumkin. Masalan, user.group yoki user:group.

- tarmoq supervizorini ishga tushirish dasturi. Agar bu maydonda kalit maydoni INTERNAL turgan bo‘lsa – bunday xizmatni inetd ning o‘zi qayta ishlaydi va yettinchi maydon kerak bo‘lmaydi.

- dasturga beriladigan parametrlar. Birinchi parametr – bu dasturning majburiy nomi.

**Tcpd dasturi.** Konfiguratsiya fayli inetd.conf faylida ko‘plab servis dasturlari nomi o‘rnida bitta va faqat tcpd dasturi ko‘rsatilgan. Bu servislarga kirishni cheklash uchun mo‘ljallangan tcp wrapper deb ataluvchi dastur hisoblanadi.

Bu dasturda ikkita konfiguratsiya fayli bor:

- **/etc/hosts.allow**
- **/etc/hosts.deny**

Birinchi faylda kim servisga bog‘lanishi mumkinligi, ikkinchisida esa kimga kirish taqiqlanganligi tavsiflangan. Fayllarda ular yuqorida qanday tartibda yozilgan bo‘lsa, shu tartibda ko‘rib chiqiladi.

Fayllarning formati bir xil:

**dastur : qaerdan [: parametrlar ]**

Birinchi maydonda servis nomi emas, dastur nomi ko'rsatiladi.

Ikkinchi maydonda – bog'lanish qaerdan mumkin (hosts.allow) yoki qaerdan taqiqlangan (hosts.deny)ligi tavsiflanadi. Bu maydonga quyidagilarni yozish mumkin:

- kompyuterlar IP adresi
- ko'rsatilgan qism tarmoqning niqobida ko'rsatilgan IP adres
- kompyuter nomi. Nomda \* belgisidan foydalanish mumkin.

Uchin maydon ba'zan qo'llaniladi va u majburiy bo'lmagan maydon hisoblanadi.

Birinchi va ikkinchi maydonlarda quyidagi kalit so'zlardan foydalanish mumkin:

- ALL – barcha dasturlar yoki tarmoqdagi xostlar
- EXCEPT – berilganlardan tashqari.

**Xinetd dasturi.** Konfiguratsiya fayli /etc/xinetd.conf da xinetd dasturi nazorat qiladigan kirish xizmatlari tavsiflanadi. Har bir xizmat quyidagicha tavsiflanadi:

```
service xizmatlar nomi
{
  parametr operator qiymat ...
  ...
}
```

Xizmatlar nomi /etc/services faylida tavsiflangan bo'lishi shart. Standart nomlardan tashqari standart bo'lmagan nomlardan ham foydalanish mumkin. Lekin bu vaziyatda port parametrini oshkor ravishda aniqlash kerak.

Quyidagi jadvalda xizmatlarni tavsiflashda qo'llash mumkin bo'lgan ba'zi parametrlar keltirilgan.

Parametrlar	Tavsifi
socket_type	Bog'lanish turi: stream (TCP), dgram (UDP), raw (strukturalashmagan), seqpacket (deytagrammalarni ketma-ket yetkazish kafolati)
protocol	Xizmatlar ishlashi bo'yicha protokollar
server	Chaqiriladigan dasturlarga to'g'ridan-to'g'ri yo'l
server_args	Dasturni chaqirishda berilishi zarur bo'lgan komandalar satri argumentlari.
port	Agar xizmat /etc/services faylida tavsiflanmagan bo'lsa, portni aniqlash shart bo'ladi.

wait	Mumkin bo'lgan qiymatlari: yes yoki no. Birinchi holda bitta yuklanadigan dastur barcha bog'lanishlarga xizmat qiladi. Ikkinchi holda esa, har bir bog'lanish uchun dasturning o'z nusxasini ishga tushirish kerak bo'ladi.
user	Dasturni ishga tushirish huquqiga ega foydalanuvchini aniqlaydi
group	Dasturni ishga tushirish huquqiga ega foydalanuvchilar guruhini aniqlaydi
nice	Yuklanuvchi dasturning nice qiymatini aniqlaydi.
access_time	Xizmatga murojaat qilish mumkin bo'lgan vaqt. Bu vaqt ss:mm-ss:mm ko'rinishida beriladi.
only_from	xinetd xizmat qiluvchi servislarga qaerdan bog'lanish mumkin bo'lgan, probel bilan ajratilgan IP adreslar yoki kompyuterlar ro'yxati.
no_access	xinetd xizmat qiluvchi servislarga qaerdan bog'lanish taqiqlangan, probel bilan ajratilgan IP adreslar yoki kompyuterlar ro'yxati.
instances	Butun son yoki UNLIMITED kalit so'zi. Bir vaqtda yuklanuvchi dasturlarning maksimal sonini aniqlaydi.
log_type	“SYSLOG vositalari” – aniqlangan vositani qo'llagan holda syslog tizimiga xabar yuborish. “FILE fayl [quyi chegara [:yuqori chegara]]” – xinetd tizimini qo'llamasdan faylga xabar yuborish. Faylning o'lchamiga chegaralar kilobaytlarda ko'rsatiladi. Quyi chegaraga yaqinlashganda xinetd xabarni generatsiya qiladi, yuqori chegaraga yaqinlashganda faylga xabarni yozishni to'xtatadi.
redirect	Bog'lanishni tarmoqdagi boshqa kompyuterga yo'naltirish. Parametr sifatida probel bilan ajratilgan IP adres va port ko'rsatiladi.
bind	So'rovni qabul qilayotgan server interfeysini aniqlaydi.
log_on_success	Muvaffiqiyatli bog'lanish holatida log fayliga qanday ma'lumot tushayotganligini aniqlaydi. Mumkin bo'lgan qiymatlari: RID, HOST, USERID, EXIT va DURATION

log_on_failure	Bog‘lanishni o‘rnatishga urinishlarda muvaffaqiyatsizlik bo‘lganda log fayliga qanday ma’lumotlar tushayotganligini aniqladi. Mumkin bo‘lgan qiymatlari: ATTEMPT, HOST, USERID, RECORD
disable	Xizmatlarni taqiqlaydi (agar qiymati yes bo‘lsa yoki ruxsat beradi (agar qiymati no bo‘lsa). Jimlik holatida kirishga ruxsat berilgan.
cps	Vaqt birligi ichida bog‘lanishlar sonini cheklaydi. Masalan, <i>cps=25 30</i> – agar bog‘lanishda bir sekund ichida 25 tadan ko‘p so‘rov bo‘lsa, xizmatga 30 sekund mobaynida kirishga ruxsat berilmaydi.
max_load	Bog‘lanishda xinetd so‘rovlarni qabul qilishni to‘xtatganda serverning maksimal yuklanishini aniqlaydi.
rlimit_*	Ruxsat berilgan dasturlarning tizimli resurslarni cheklashga yordam qiluvchi turli parametrlar. rlimit_as, rlimit_cpu, rlimit_data, rlimit_rss, rlimit_stack

Konfiguratsiya faylida xizmatlarni tavsiflashda quyidagi atributlarni albatta ko‘rsatish shart:

- **socket\_type**
- **wait**
- **user**
- **server**
- **protocol**

Konfiguratsiya faylida asosiy konfiguratsiya fayliga qo‘shilishi kerak bo‘lgan katologni aniqlashda yordam beradigan include parametridan foydalaish mumkin.

Quyida /etc/xinetd.conf faylining mazmuni keltirilgan:

```
defaults
{
instances          = 60
log_type           = SYSLOG authpriv
log_on_success     = HOST PID
log_on_failure     = HOST
cps                = 25 30
}
```

**includerdir /etc/xinetd.d**

defaults – jimlik holati bo'yicha barcha servislar uchun parametrlarni aniqlaydi.

Qo'shimcha konfiguratsiya fayllari /etc/xinetd.d katologida joylashadi.

Ror3 servisini tavsiflovchi /etc/xinetd.d/popa3d fayliga misol:

```
service pop3
{
    socket_type      = stream
    wait              = no
    user              = root
    server           = /usr/sbin/popa3d
    log_on_success += USERID
    log_on_failure += USERID
    disable          = yes
}
```



1. Qanday parametr yordamida har bir yangi bog'lanish uchun dasturlarning yangi nusxasini ishga tushiruvchi tarmoqning superserverini ko'rsatish mumkin?
2. Qanday qilib, aniq kompyuterning dasturga kirishini cheklash mumkin?
3. Siz xizmatdan foydalanishga ruxsat berdingiz. Qanday qilib tarmoq superserveriga o'zining konfiguratsiya faylini o'qishini ta'minlash mumkin?



## 5-BOB. DNS SERVERNI SOZLASH

### 5.1-§. Domain Name System

Tarmoqda kompyuterlar soni oshib borishi bilan ularga IP-lar orqali murojaat qilish noqulayliklar tug'dira boshladi, shuning uchun ham tarmoqdagi kompyuterlarga alohida nom qo'yish boshlandi. Dastlab kompyuter nomini IP adresga akslantirish va teskari akslantirish uchun /etc/hosts fayli ishlatilgan. Bu fayl yordamida nomni to'g'ri akslantirish shartlaridan biri uning tarmoqdagi barcha kompyuterlarda bir xil mazmunda bo'lishi hisoblanadi.

Tarmoqda bir necha yuzlab kompyuterlar bo'lsa, ular o'rtasida hosts faylini sinxronlash hech qanday muammo tug'dirmaydi. Lekin, agar ayni vaqtda Internet-ga ulangan kompyuterlar sonini tassavur qila olsangiz, u holda hosts faylini ishlatish juda ko'p qiyinchiliklarga olib keladi. Agar tarmoqqa yangi kompyuter ulansa, bu faylni tarmoqdagi barcha kompyuterlarda almashtirib chiqish kerakligini tasavvur qiling. Bundan tashqari, bu faylning o'lchamini hisobga olsak, har bir kompyuterda bitta satr qo'shilishi kerak bo'ladi. Ya'ni, /etc/hosts faylini akslantirish uchun ma'lumot manbai sifatida ishlatadigan bo'lsa, u holda tarmoqdagi barcha trafik ushbu faylni sinxronlash uchun sarflanadi.

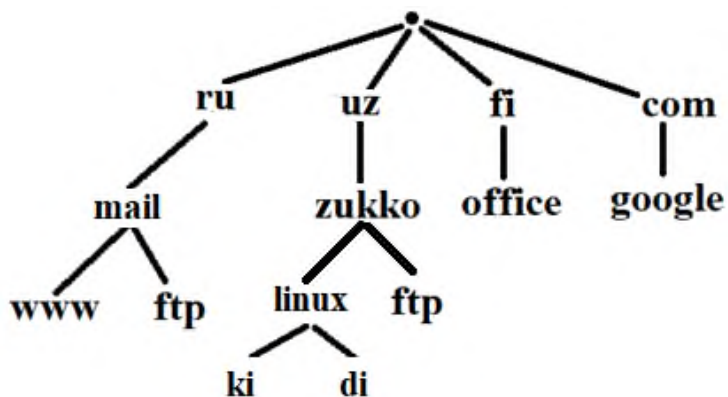
20-asrning 80-yillari boshida DNS (-Domain Name System) tizimi ishlab chiqildi. Bu tizim IP adresga kompyuter nomlarini akslantirish va teskari akslantirish uchun mo'ljallangan, taqsimlanuvchi ma'lumotlar bazasi sifatida taqdim etiladi. DNS tizimi hosts faylining kamchiliklarini bartaraf etadi va nomlarni akslantirishni tezkorlik bilan amalga oshiradi.

**DNS nomi maydoni.** DNS-da domen tushunchasi kiritilgan. Tarmoqda joylashgan ixtiyoriy kompyuter DNS ning alohida domeni bilan aniqlanadi.

DNS domeni va Windows domenlarini adashtirib yubormaslik kerak. Ular turli xil maqsadlarda qo'llaniladi. DNS domeni faqat nomlarni akslantirish uchun qabul qilingan bo'lsa, **Windows domenlari xavfsizlik masalalari bilan bog'liq.**

DNS-ning domen strukturasi ierarxik ko'rinishga ega. Nomlar ierarxiyasi “.” belgisi bilan ajratib ko'rsatiluvchi ildiz domendan boshlanadi. Ildiz domenda birinchi darajali domenlar joylashadi. Ular geografik tamoyillar asosida ishlab chiqilgan bo'lishi mumkin: .ru,

.uz, ua, .uk va boshq. Shu bilan birgalikda tarixiy tamoyillar bo'yicha ham ishlab chiqiladi, ya'ni, bunday domenlar dastlab Amerika Qo'shma shtatlarida qo'llanilgan: com, net, mil va boshq. Ildiz domenlardan keyin ikkinchi, uchinchi va h.k. darajadagi domenlar yoziladi.



DNS da to'liq kvalifikatsiyalangan domen nomlari tushunchasi mavjud - FQDN. Bunday nomlar ikki qismdan iborat bo'ladi: kompyuter nomi va domen nomi.

Masalan:

- www.zukko.uz – www kompyuterlari nomi va zukko.uz domen nomidan tashkil topgan.

- c1.of.zukko.uz – kompyuter nomi c1 va domen nomi of.zukko.uz .

Misollardan ko'rinib turibdiki, birinchi nom – bu kompyuter nomi bo'ladi. Qolgan qismi esa domen nomi bo'ladi.

## 5.2-§. DNS serverning javobgarlik hududi (sohasi)

Agar Internetda har bir DNS server tarmoqdagi barcha kompyuterlarga javob beradi deb faraz qilsak, u holda bu hosts faylini qo'llashdan farq qilmay qoladi. Shuning uchun ham DNS taqsimlanuvchi ma'lumotlar bazasi ko'rinishida bajariladi.

Har bir DNS serverning o'z javobgarlik hududi mavjud. U Internet tarmog'iga to'liq javob bermaydi, u faqat Internetning kichik bir qismiga javobgarlikni o'z bo'yniga oladi. Javobgarlik hududi – bu DNS server nomlarini IP adreslarga akslantiruvchi aniq domenga ega bo'lgan kompyuterlar hisoblanadi.

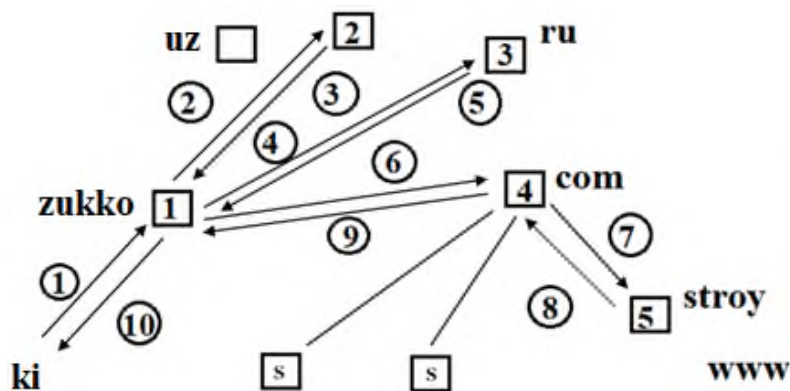
Masalan, zukko.uz domeniga javob beruvchi DNS server, ushbu domenga kiruvchi kompyuterlar nomini akslantirishga mo'ljallangan. Agar oldingi mavzuda berilgan sxemaga e'tibor bersak, DNS server

quyidagi kompyuterlarga ham javob berishini tushunib olamiz: office.zukko.uz, www.zukko.uz, ftp.zukko.uz . Keyingi darajadagi c1.office.zukko.uz , c2.office.zukko.uz domenlar kompyuterlari boshqa domenga tegishli bo‘lganligi uchun, yuqoridagi DNS serverom ularning nomlarini akslantirmaydi.

### 5.3-§. DNS-ning ishlash prinsiplari.

DNS-ning ishlash prinsiplari uchun yuqorida ko‘rsatilgan sxema qo‘llaniladi. Faraz qilaylik, barcha DNS serverlar bir vaqtda ishga tushirilgan (qo‘shilgan) bo‘lsin va birorta ham so‘rovlarga javob bermasin. Aniq bir domenga javob beruvchi DNS server (sxemada kvadrat bilan belgilangan), ushbu domen bilan kompyuterda joylashgan bo‘lsin. Real hayotda DNS server bir vaqtda bir nechta domenlarni qo‘llaydi va tarmoqning ixtiyoriy joyida (o‘rnida) joylashgan bo‘lishi mumkin.

ki.zukko.uz kompyuteridagi brauzerda www.stroy.com.ru-ni terib Enter tugmasini bosamiz. Buning uchun brauzer mos keluvchi WEB serverga ushbu serverning IP adresini olish uchun paketni yuboradi. U shu kompyuterda joylashgan DNS klientiga murojaat qiladi, klient esa o‘z navbatida DNS server o‘rnatilgan kompyuterga so‘rov yuboradi.



Klientga xizmat ko‘rsatuvchi DNS server zukko.uz kompyuterida joylashgan. Aynan shu serverga klient birinchi so‘rovni yuborgan (sxemadagi 1-qadam). Ya’ni www.stroy.com.ru kompyuteri zukko.uz-ning DNS serveri javob beradigan hududga kirmaydi, shuning uchun ham bu server klientning so‘roviga javob bera olmaydi va ildiz (ildiz) domenga javob beruvchi DNS serverga murojaat qiladi (sxemadagi 2-qadam).

Ildiz DNS serveri ham stroy.com.ru kompyuteriga javobgar

emas, shuning uchun yakunlanuvchi javobni bera olmaydi. Lekin ushbu server .ru domeniga javobgar DNS server nomi va IP adresini biladi. Shuning uchun ham ushbu ma'lumotlarni bizning serverga javob sifatida qaytaradi (sxemadagi 3-qadam).

Keyingi so'rov .ru domeniga javobgar bo'lgan DNS serverga yuboriladi (sxemadagi 4-qadam). Lekin bu server ham klientning so'roviga javob bera olmaydi, chunki bu server ham stroy.com.ru domeni uchun javobgar emas. Lekin ushbu server com.ru domeniga javobgar bo'lgan DNS serverning nomi va IP adresini biladi va bizga ushbu qiymatlarni qaytaradi (sxemadagi 5-qadam).

Bizning DNS server com.ru domeniga javobgar bo'lgan DNS serverga so'rov yuboradi (sxemadagi 6-qadam). Bu DNS server ham stroy.com.ru domeniga javobgar emas, lekin ushbu domenga javobgar bo'lgan DNS serverning nomi va IP adresini biladi hamda mustaqil ravishda ushbu serverga so'rov yuboradi (sxemadagi 7-qadam). Nima uchun bu klient yuborgan so'rov bo'yicha ma'lumotlarni mustaqil qidirishi haqida keyinroq so'z yuritamiz.

stroy.com.ru domeniga javob beruvchi DNS server www kompyuterning IP adresini qaytaradi (sxemadagi 8-qadam). Ushbu IP adres bizning DNS serverga qaytib keladi (9-qadam) va uni klientga yuboradi (10-qadam).

#### **5.4-§. Keshlovchi DNS server**

Statika bo'yicha bitta WEB sahifada 15 ta atrofida kartinka joylashgan bo'ladi. Agar HTTP protoklining 1.1. versiyasini qo'llay olmaydigan WEB brauzerdan foydalansak, u holda har bir kartinka uchun alohida so'rov shakllantiriladi. Har bir so'rovda bizning DNS serverga 15 ga yaqin so'rov va IP adresni aniqlash protsedurasi takrorlanadi. Agar so'rovlar aynan shunday takrorlansa, taqdim etilgan Internet trafigining asosiy qismi DNS so'rovlarni qayta ishlashga sarflanib ketadi.

Buning oldini olish maqsadida, ya'ni takrorlanuvchi so'rovlar muammosini yechish uchun barcha DNS serverlar keshlovchi hisoblanadi. Bizning DNS server lokal keshida www.stroy.com.ru haqidagi ma'lumotlar joylashadi va keyingi so'rov tashkil etilganda ma'lumotlar ushbu keshdan olib beriladi. Bundan tashqari bu ma'lumotlar so'rovlarga javob qaytaruvchi sxemadagi 2, 3 va 4 raqamlari bilan belgilangan DNS serverlarda ham saqlanib qoladi.

Server keshidagi ma'lumotlarning saqlanish vaqti hudud sozlamalariga bog'liq bo'ladi. Ko'p hollarda bu vaqt bir sutka atrofida bo'ladi.

**Rekursiv va rekursiv bo'lmagan DNS serverlar.** Agar keltirilgan sxemadagi 2, 3 va 4 DNS serverlar faoliyatiga e'tibor qaratsak, ulardagi farqlarni sezishimiz mumkin. Dastlabki ikkita server DNS serverlar ierarxiyasida quyida turuvchilar haqida ma'lumot beradi. 4-server esa 5-serverga ma'lumotlar yuboradi.

Bu ikkala server rekursiv bo'lmagan serverlar, 4-server esa rekursiv server hisoblanadi. Rekursiv serverlar qiziqtirgan ma'lumotlar bo'yicha barcha qadamlarni mustaqil bajarishga harakat qiladi, rekursiv bo'lmagan serverlar esa faqat o'zlari javobgar bo'lgan hududlar yoki ularning keshida saqlanayotganlar haqida ma'lumot beradi.

Birinchi va ikkinchi darajali domenlarga javobgar bo'lgan DNS serverlar odatda eng ko'p so'rovlarni qayta ishlasa ham rekursiv bo'lmagan server hisoblanadi. Masalan, ildiz domenga javobgar bo'lgan DNS serverlar bir sekundda 10000 atrofidagi so'rovlarni qayta ishlaydi.

Sxemadagi 4-raqam bilan belgilangan DNS serverni ham rekursiv bo'lmagan server deb hisoblash mumkin. DNS serverning rekursiv yoki rekursiv bo'lmasligi, uni qanday sozlanganligiga bog'liq bo'ladi.



*DNS klientlari so'rovlariga xizmat ko'rsatuvchi DNS serverlari albatta klient so'rovlari uchun rekursiv bo'lishi kerak. Chunki, DNS klientlari boshqa DNS serverlariga jo'natmalarni (so'rovlarni) bajarishni bilmaydi.*

Biz ko'rayotgan misoldagi DNS serverlar ichki DNS klientlariga rekursiv server sifatida ko'rinadi, tashqi so'rovlar uchun esa rekursiv bo'lmagan server sifatida bo'ladi.

**Nufuzli (ahamiyatli) va nufuzli bo'lmagan javoblar.** Faraz qilamiz, stroy.com.ru domeni uchun javobgar bo'lgan DNS serveri qandaydir sababga ko'ra kirishga ruxsati yopiq bo'lsin. U holda 4-raqamli DNS serveri www.stroy.com.ru domeni kompyuteri haqida ma'lumotlarni ololmaydi va ma'lum bir vaqt o'tgandan keyin berilgan so'rovga www.stroy.com.ru kompyuteri mavjud emas degan javob keladi. Bu ma'lumot bizning serverning keshiga klientga javob berish

uchun kelib tushadi.

Endi faraz qilaylik, ma'lum bir vaqt oralig'ida stroy.com.ru domeni uchun javobgar bo'lgan DNS serveri ishchi holatga keladi, ya'ni kirishga ruxsat tiklanadi va so'rovlarga javob berishni boshlaydi. Lekin bizning serverning keshida oldindan salbiy javob haqidagi ma'lumot saqlanib turibdi, o'rtacha holda bir sutka davomida serverimiz klientga salbiy javobni uzatib turaveradi.

Bunday muammoli vaziyatlardan chiqish uchun o'z domenimizga xizmat ko'rsatuvchi qo'shimcha DNS serverdan foydalanish kerak bo'ladi. Sxemada bunday qo'shimcha serverlar S belgisi bilan belgilangan. 4-serverga stroy.com.ru domeni uchun javobgar bo'lgan barcha DNS serverlar ma'lum, agar 5-server so'rovlarga javob bermasa, 4-server ushbu domen uchun mas'ul bo'lgan boshqa DNS serverga murojaat qiladi.

5-server master DNS serveri deb ataladi. Qo'shimcha serverlar esa bo'ysinuvchi (slave) DNS serverlar deyiladi. Domendagi barcha o'zgarishlar faqat master serverida amalga oshirib boriladi, slave serveridagi ma'lumotlar bevosita tahrirlanmaydi, ular ma'lumotlarni master serveridan oladi.

DNS klienti ma'lumotlarni nufuzli serverlar - master yoki slave dan olsa, bu javob ma'lumotlari nufuzli deb ataladi. Agar ma'lumot keshdan olinsa, u holda bunday javob nufuzli bo'lmagan javob deyiladi.



1. mail.ru kompyuteri qanday domenga tegishli?
2. Bir vaqtda bir nechta domenlarga mas'ul bo'lgan DNS server bo'lishi mumkinmi?
3. Rekursiv bo'lmagan DNS server bilan ishlovchi DNS klient bo'lishi mumkinmi?
4. Domenga va ichki klientlarga xizmat ko'rsatuvchi tashkilotning DNS serverini yarating. Ushbu server tashqi so'rovlar uchun rekursiv bo'lmagan, ichki klientlar uchun rekursiv bo'lishi mumkinmi?
5. Sizing hududingizni qo'llab turuvchi slave serveri albatta bo'lishi shartmi?

## 5.5-§. BIND DNS serveri

BIND (Berkeley Internet Name Domain) serveri - UNIX da ancha ommalashgan DNS serveri hisoblanadi. Bu server ISC (Internet Software Consortium) tashkiloti tomonidan dastur kodi bilan tarqatiladi. Linux-ning barcha distributivlarida qo'yilgan, jumladan ASPLinux-da ham mavjud.

BIND 1985 yilda Kevin Danlap (Kevin Dunlap) tomonidan ishlab chiqilgan. Hozirgi paytda uning 9-versiyasidan foydalaniladi. U DNS tizimining barcha asosiy xususiyatlarini mujassamlashtirgan, jumladan, oxirgi paytlardan qo'shilgan quyidagi barcha yangiliklarni o'zida birlashtiradi:

- Ipv6 ni qo'llash;
- DNSSEC
- TSIG
- DNS kengaytirilgan - EDNSo protokollarini qo'llash.

Asosan BIND serverining EDNSo-ni qo'llashini alohida ta'kidlash kerak. Kengaytirilgan protokolda TCR protokolining nafaqat hududni uzatish uchun imkoniyatlari, balki so'rovlarni raqamli imzo bilan yuborish imkoniyati ham qo'shilgan. Bundan tashqari, hududni inkrementli yuborish va dinamik yangilashni qo'llab quvvatlaydi (DHCP serverlari uchun).

**Domenni qo'llashni yoqish.** BIND serverining konfiguratsiya fayllari.

O'qituvchining DNS serverida linux.zukko.uz domenini qo'llab-quvvatlovchi serverni yaratish.

Keyingi o'rinlarda BIND DNS serverini o'rganishni quyidagicha amalga oshiramiz:

Dastlab, o'qituvchi linux.zukko.uz domenini qanday qilib qo'llab-quvvatlovchi serverni hosil qilish mumkinligini tushuntiradi. Domen o'qituvchining DNS serverida hosil qilinadi. Buning uchun barcha konfiguratsiya fayllari va bu fayllarga kiritiladigan o'zgarishlar ko'rsatib o'tiladi.

Xuddi shunday, DNS serverning domenlar bilan ishlovchi named dasturi ishga tushirilib, konfiguratsiya fayllaridagi sintaksisni tekshirish ko'rsatiladi.

Shundan keyin tinglovchilarga linux.zukko.uz domeniga qismdomen ochish huquqi beriladi va ular mustaqil ravishda o'zlarining serverlarida domenni qo'llab-quvvatlovchi DNS serverni

hosil qilishni laboratoriya ishi sifatida bajaradi.

DNS serverlarni tinglovchilar kompyuterlarida ishga tushirgandan keyin, BIND DNS serverini turli holatlarda qo'llashning turli jihatlari tushuntirib beriladi.

### 5.6-§. BIND serverining konfiguratsiya fayllari

BIND serverining asosiy konfiguratsiyai fayli /etc/named.conf fayli hisoblanadi.

named.conf faylidan tashqari qo'shimcha konfiguratsiya fayllaridan ham foydalaniladi:

- hudud tavsifi
- kalitlar tavsifi
- maslahat fayllari.

**named.conf fayli.** DNS serverning asosiy konfiguratsiya parametrlari va ushbu server qo'llab-quvvatlaydigan hudud parametrlari aniqlanadi.

Faylda qo'llaniladigan ko'rsatmalar:

- include
- options
- acl
- zone
- view va boshqalar.

named.conf faylida DNS serverning asosiy konfiguratsiya parametrlari va ushbu server qo'llab-quvvatlaydigan hudud parametrlari aniqlanadi. Bu faylda C/C++ va shell script tillaridagi usullar bilan izohlarni qo'llash mumkin:

/\* Bu izoh bir nechta satrlarda joylashishi mumkin \*/

// Bu izoh bitta satrda joylashgan bo'ladi

# Bu izoh bitta satrda joylashgan bo'ladi

Bu fayl ko'rsatmalardan iborat. Har bir ko'rsatma uning turini aniqlovchi kalit so'zlari bilan boshlanadi va albatta «;» belgisi bilan tugallanishi shart. Agar ko'rsatma parametrlari bitta satrga sig'masa, u holda bunday ko'rsatmalarni figurali qavslar { va } oralig'iga olib yozish kerak bo'ladi.

Quyidagi jadvalda ko'rsatmalarda qo'llaniladigan so'zlar ro'yxati keltirilgan:

Ko'rsatma	Tavsifi
include	Tashqi faylni qo'shish



options	BIND serverining global parametrlarini aniqlash
server	Server parametrlarini kiritish
lwres	BIND 9 serverini va soddalashtirilgan tanib olish xususiyatlarni sozlash
key	Autentifikatsiya parametrlarini aniqlash
acl	Boshqaruv ro'yxatini aniqlash
zone	Server qo'llab-quvvatlovchi hududni aniqlash
trusted-keys	Konfiguratsiya faylida aniqlanadigan shifrlash kaliti
controls	BIND serverini rndc utiliti qanday boshqarishini aniqlash
logging	Log xabarleri toifasi va ularni tarqatish kanallarini aniqlash
view	Nomlar maydoni taqdim etishni aniqlash

Ko'rsatmada kompyuter va tarmoqning IP adreslarini tavsiflash zarur bo'ladi. Quyida adreslarni yozishning mumkin bo'lgan variantlari keltirilgan:

- IP adres. Masalan, 192.168.0.0.
- Qismtarmoq niqobi bilan tarmoq adresi, masalan, 192.168.0.0/24.
- Kirishga ruxsat ro'yxatida oldindan aniqlangan nom (acl).
- inkor operatori - "!".

**Options ko'rsatmasi.** Bu ko'rsatma yordamida BIND serverining global parametrlari beriladi.

```
options {
  parametrlar;
  parametrlar;
}
```

Quyidagi options ko'rsatmasida aniqlanadigan ba'zi parametrlar keltirilgan:

Parametrlar	Tavsifi
directory	BIND serverning hududni tavsiflovchi fayllari va turli xil qo'shimcha fayllarni hosil qilish uchun katologni aniqlash. Odatda bu parametrlar /var/named katologiga yo'naltiriladi.

	Masalan: directory “/var/named”
notify yes no	Agar notify parametri yes-ga teng bo‘lsa, u holda hudud tavsifi o‘zgartirilganda xabarlar barcha slave DNS serverlariga uzatiladi. Masalan: notify no
also-notify address	Bu parametr yordamida hudud tavsifi o‘zgartirilganligi xabari beriladigan DNS server aniqlanadi. Bu ro‘yxatda Slave DNS serverini ko‘rsatish shart emas. Jimlik holatida qiymatlar aniqlanmagan. Masalan: also-notify { 193.12.20.1; 193.12.38.200; };
recursion yes no	Bu parametr yordamida BIND serveri rekursiv serverligi aniqlanadi. Jimlik holatidagi qiymat – yes. Masalan: recursion yes { 193.12.13.240; 194.12.34/24; };
allow-recursion address	Bu parametr BIND serveri rekursiv server sifatida ko‘rinadigan tarmoq yoki kompyuterlar adresini aniqlaydi. Jimlik holatida qiymatlari aniqlanmagan. Masalan: allow-recursion
listen-on port list	Bu parametr qaysi interfeys va portlar server so‘rovlariga bo‘ysinishini aniqlaydi. Jimlik holatidagi qiymatlari: barcha interfeyslar va 53 porti. Masalan: listen-on { 5.6.7.8; }; listen-on port 1234 { !195.158.1.1; 1.2/16; };
allow-query adres	Bu parametr bizning serverga qaysi adreslardan so‘rovlar yuborishini aniqlaydi. Jimlik holatidagi qiymat: barcha adreslarga ruxsat. Masalan: allow-query { 195.158.1.1; 172.20.20/24; };
allows transfer	Bu parametr qaysi serverlar uchun hududiy

adres	junatmalarga ruxsatni aniqlaydi. Jimlik holatidagi qiymat: jo'natmalar uchun barcha serverlarga ruxsat. Masalan: allows transfer { 195.158.1.1; 172.20.20/24; };
blackhole adres	Bu parametr yordamida hamma vaqt so'rovlari inkor qilinadigan serverlar aniqlanadi. Server bu serverlarga o'zining jo'natmalarini uzatmaydi. Jimlik holatidagi qiymati: ro'yxat aniqlanmagan. Masalan: blackhole { 195.158.1.1; 2.3.4.5; };

**ACL instruktsiyasi.** Asl – kirishga ruxsatni boshqarish ro'yxati. acl instruktsiyasi yordamida tez-tez qo'llaniladigan IP adreslarni aniqlash va unga keyinchalik boshqa ko'rsatmalarda ko'rsatkich qo'yish uchun nom berish mumkin.

```
acl nom {
  ro'yxat elementlari;
};
```

acl uning nomini boshqa parametrlarda qo'llaniladigan holat yuz berguncha aniqlanishi kerak. Oldindan aniqlangan to'rtta ro'yxat mavjud:

- any – barcha tugunlarga mos keladi.
- localnets – lokal tarmoqning barcha tugunlarini mos keladi.
- localhost – kompyuterning o'ziga mos keladi.
- none – birorta ham tugunga mos kelmaydi.

Acl-ni aniqlashga misol:

```
acl internal { 195.158.1.1; 2.3.4.5;
172.20.20/24; 192.168.0/24; };
```

**Zone ko'rsatmasi.** DNS server mas'ul bo'lgan hududni tavsiflash uchun mo'ljallangan. DNS serveri qo'llab-quvvatlaydigan har bir domen uchun alohida zone ko'rsatmasi yozilishi kerak.

Bu ko'rsatmada qo'llaniladigan parametrlar hudud turiga bog'liq. Hudud turi type parametri yordamida aniqlanadi.

Hududni tavsiflashga misol:

```
zone "domen nomi" IN {
  type hudud turi;
```

```

    hudud parametrlari;
};

```

Quyidagi jadvalda mumkin bo‘lgan hudud turlari berilgan:

Turi	Tavsifi
master	Asosiy (master) hududni aniqlaydi
slave	Bo‘ysinuvchi (slave) hududni aniqlaydi
hint	Masalahat hududni belgilaydi
forward	Qayta adreslash (o‘tkazish) hududini aniqlaydi
stub	Kelajakda foydalanish uchun tavsiya etilmaydi

Yuqorida sanab o‘tilgan hududlarni tavsiflovchi parametrlar BIND DNS serverining domenlarni qo‘llab quvvatlashini qurish usullarini tavsiflashda qarab chiqamiz.

### 5.7-§. Hududlarni tavsiflash fayli

Aniq domen haqidagi asosiy ma’lumotlar hududlarni tavsiflash faylida joylashadi. Bu faylda “yozuv” deb ataluvchi resurslar tavsiflanadi, bular RFC da aniqlanadi: 882, 1035, 1183, 2065, 2181, 2308 va 2535.

Hududni tavsiflash faylida quyidagi maxsus belgilardan foydalanish mumkin:

- ; - izoh.

- @ - joriy domen nomi. Bu nom zone ko‘rsatmasidan yoki hududni tavsiflash faylining \$ORIGIN direktividan olinadi.

- O – ma’lumotlarni bir nechta qatorga bo‘lish.

- \* - faqat kompyuter yoki domen nomida qo‘llaniladi.

**Direktivlar.** Hududni tavsiflash faylida maxsus direktivlardan foydalanish mumkin:

- \$TTL vaqt – barcha hudud yozuvlari uchun DNS serverning keshidagi yozuvning hayotiy tsiklini aniqlaydi.

- \$ORIGIN domen - @ belgisi bilan almashtirilgan domen nomini aniqlaydi yoki to‘liq aniqlanmagan kompyuter nomi yoki domenning oxiriga almashtiriladi.

- \$INCLUDE fayl – tashqi faylni qo‘shish (chaqirish).

- \$GENERATE parametrlar – o‘xshash yozuvlar hosil qilish

uchun mo'ljallangan.

**Hududlarni tavsiflash faylida FQDN nomining yozilish xususiyatlari.** To'liq kvalifikatsiyalangan yoki to'liq domen nomi (FQDN)ni ko'rsatuvchi kompyuterda faqat va faqat hududni tavsiflash faylida ildiz domenning nomini aynan yozish talab etiladi.

Masalan, agar master.linux.zukko.uz kompyuteridagi hudud tavsiflash faylida FQDN nomini ko'rsatish zarur bo'lsa, uni quyidagi namunada tavsiflash kerak:

**master.linux.zukko.uz .**

Nomdan keyin qo'yilgan nuqtaga e'tibor berish kerak – bu ildiz domenga aniq ko'rsatkich hisoblanadi.

Agar oxiridagi nuqta qo'yilmasa, u holda DNS server avtomatik ravishda joriy domen nomini qo'yadi (joriy domen nomi yoki zone ko'rsatmasidan yoki named.conf konfiguratsiya faylidan yoki \$ORIGIN direktividan olinadi). Masalan, joriy domen linux.zukko.uz bo'lsa va hududni tavsiflash faylidan bu nom nuqtasiz master.linux.zukko.uz kabi yozilgan bo'lsa, u holda DNS server quyidagi qiymatni qo'yadi:

**master.linux.zukko.uz .linux.zukko.uz .**



*Hududni tavsiflash faylini yozishda eng ko'p uchraydigan xatolik – bu FQDN nomidan keyin nuqtani qo'ymaslik hisoblanadi.*

## 5.8-§. Resurslar haqida yozuv

Yozuvning formatini quyidagi namuna bo'yicha taqdim etish mumkin:

**[kompyuter nomi|domen nomi] [ttl] [klass]  
parametr turi**

Birinchisi – bu domen yoki kompyuter nomi. Bu maydon aynan yozish, yozuv turiga bog'liq bo'ladi. Agar bu maydon bo'sh qoldirilsa, uning qiymati oldingi yozuvlardan olinadi.



*Agar birinchi maydonni ko'rsatmasak, u holda qator boshida yoki probel yoki tabulyatsiya belgisini qo'yish shart.*

Ikkinchi maydon - DNS server keshidagi yozuvlarning hayotiy vaqti. Maydonning qiymati sekundlarda o'rnatiladi. Maydon qiymati bo'sh bo'lsa, uning qiymati berilgan hudud uchun jimlik holatida

aniqlanadi.

Uchinchi maydon – tarmoq klassi. Quyidagi tarmoq klasslarini qo‘llash mumkin:

- IN - Internet (jimlik holatidagi qiymat).
- CN – ChaosNet. Hozirgi paytda ishlatilmaydi.
- HS - Hesoid – BIND paketiga qo‘shimcha hisoblanuvchi axborot xizmati. Juda kamdan-kam hollarda qo‘llaniladi.

Yozuv turi – zahiradagi so‘z. Jadvalda asosiy yozuv turlari keltirilgan:

<b>Tur</b>	<b>Tavsifi</b>
SOA	DNS hududi parametrlarini aniqlash. Majburiy yozuv.
NS	Hududdagi nufuzli DNS serverlarni aniqlash. Qism domenlar vakili. Majburiy yozuv.
A	IP adresga nomni akslantirish
AAAA	IPv6 adresga nomni akslantirish
A6	IPv6 adresga nomni akslantirish
PTR	IP adresni nomga akslantirish
MX	Pochta domeniga mas’ul bo‘lgan pochta serveri nomini ko‘rsatish uchun qo‘llaniladi.
KEY	DNS nomi uchun shifrlashning ochiq kaliti
CNAME	Kompyuterning qo‘shimcha nomi (psevdo nom)
SRV	Domenlararo xizmatlarni aniqlash

Yozuvlarning ba’zi boshqa turlarini qo‘llash mumkin.

Ixtiyoriy hududni tavsiflash faylida quyidagi ikkita yozuvni aniqlash shart: SOA va NS.

**SOA yozuvi.** SOA (Start Of Authority) yozuvi – hududni tavsiflashning boshini aniqlaydi. Bu majburiy bo‘lgan yozuvlardan biri hisoblanadi. Bu hududni tavsiflash faylining eng birinchi o‘rinda bo‘lishi shart. Fayldagi hududni tavsiflash boshqa SOA yozuvi uchramaguncha davom etadi.

SOA yozuvining formatiga misol:

```
linux.zukko.uz . IN SOA
master.linear.zukko.uz . {
    ilhom.linear.zukko.uz . ; e-mail
    2016032200 ; yozuvning seriya nomeri
    18H ; yangilash vaqti
    20M ; o‘rinishlar orasidagi vaqt
    2W ; eskirish intervali
```

## 10 } ; TTL

SOA yozuvi qanday maydonlardan tashkil topganligini ko'rib chiqamiz.

Birinchi parametr – bu domen nomi. Bu parametrda joriy domen nomi o'rniga qo'yiladigan @ - maxsus belgisidan foydalanish mumkin.

Ikkinchi (shart bo'lmagan) TTL parametri ko'rsatilmagan, shuning uchun ham bu yozuvning DNS serveri keshidagi saqlanish vaqti ushbu huddad uchun jimlik holatidagi qiymatdan olinadi.

Uchinchi parametr - tarmoq klassi IN sifatida aniqlanagan. Shunga qaramasdan bu parametr majburiy emas, bu parametr hudud tavsifi faylining o'qilishini yaxshilash uchun aniqlanishi tavsiya etiladi.

Keyin SOA yozuvi turi qo'yilgan. Qolgan barcha parametrlar SOA turidagi yozuvlarga xos bo'lgan parametrlar hisoblanadi.

Dastlab ushbu hududga mas'ul bo'lgan master DNS serverining nomi ko'rsatiladi.

Ikkinchi parametr – ushbu hududga javobgar shaxsning pochta adresi. Pochta adresi ixtiyoriy domenda joylashgan bo'lishi mumkin.

2016032200 – hudud yozuvining seriya nomeri. Hududni tavsiflash faylini o'zgartirishda ushbu sonni kattalashtirishga majbursiz, chunki bu nomer hududiy jo'natmalarni aniqlash zarurati uchun slave DNS serverlarida qo'llaniladi. Agar slave serverning seriya nomeri master DNS serverining seriya nomeridan katta yoki teng bo'lsa, hududiy jo'natmalar bo'lmaydi. Agar seriya raqami kichik bo'lsa, slave serverga hududiy jo'natmalar amalga oshadi.

Nomer sifatida 1, 2, 3, 4 va boshqalarni ishlatish mumkin. Lekin, real hayotda seriya raqami sifatida hududni tavsiflovchi faylda amalga oshirilgan oxirgi o'zgartirish sanasini qo'yish tavsiya etiladi. Berilgan misolda faylga oxirgi o'zgartirish 2016 yil mart oyining 22 sanasida kiritilgan. Oxirgi 00 raqamlari aynan shu kunda faqat bitta o'zgarish bo'lganligini anglatadi.



*Ushbu maydondagi belgilarning maksimal soni 10 ga teng deb olingan.*

Keyin, slave serverning master serverga maydondagi seriya nomerlarini taqqoslash uchun qilinadigan murojaat intervalini aniqlovchi vaqt keltirilgan.



*SOA turidagi yozuv parametrlarida vaqt intervali sekundlarda beriladi. Lekin BIND serveri vaqtni aniqlashning qisqa variantini qoʻllaydi.*

Agar slave server master serveriga murojaat vaqti kichik intervalida bogʻlana olmasa, u boshqa ish rejimiga oʻtadi. 20M parametri ushbu intervalni aniqlaydi.

Slave server uzluksiz ravishda urinishlardan keyin ham master serveriga ulana olmasligi mumkin, keyingi 2W parametri slave serveri bu hududni qoʻllab-quvvatlashdan voz kechish vaqtini aniqlaydi.

Oxirgi parametr jimlik holati boʻyicha DNS serverning keshidagi salbiy maʼlumotning saqlanish vaqtini aniqlaydi.

SOA turidagi yozuvning barcha parametrlari majburiy hisoblanadi.

**NS yozuvi.** NS (Name Server) turidagi yozuv berilgan domenlar uchun nufuzli boʻlgan barcha DNS serverlarni tavsiflash uchun moʻljallangan. Ushbu yozuv yordamida barcha master va slave serverlarni tavsiflash kerak. Shu bilan birgalikda bu yozuv hududdagi vakolatli huquqni qabul qiladi.

NS yozuvi majburiy yozuvlardan hisoblanadi. Hududni tavsiflash faylida hech boʻlmaganda bitta shunday yozuv boʻlishi kerak.

NS yozuvining formati:

**[hudud] [TTL] [IN] NS server nomi**

Agar linux.zukko.uz hududiga master.linear.zukko.uz va moodle.samtuit.uz DNS serverlari javob bersa, u holda hududni tavsiflovchi faylga quyidagi ikkita yozuvni kiritish kerak boʻladi:

**@ IN NS master.linear.zukko.uz .**

**@ IN NS moodle.samtuit.uz .**

Yuqorida sanab oʻtilganlardan qaysi biri master, qaysi biri slave server hisoblanishi SOA yozuvlarining birinchi parametrlari yordamida aniqlanadi.

**MX yozuvi.** MX (Mail Exchanger) turidagi yozuvlar domen uchun pochta qabul qilishga javobgar boʻlgan pochta serverlarini koʻrsatish uchun moʻljallangan.

MX yozuvining formati:



## **[domen] [TTL] [IN] MX pochta serverining nufuzi**

Pochta tizimini yaratish aynan DNS-ni ishlab chiqishga o'xshash bo'lib, asosiy va yordamchi pochta serverlari ajratiladi. Jo'natma uzatish nufuzi «nufuz» maydoni yordamida aniqlanadi. Ushbu maydondagi son qancha kichik bo'lsa, ko'rsatilgan serverning nufuzidan shuncha katta bo'ladi. Ya'ni, jimlik holatida jo'natma pochta serveriga katta nufuz bilan uzatiladi. Biroq qandaydir sabablarga ko'ra pochta serverida kirishga ruxsat yopiq bo'lsa, jo'natma serverga kichik nufuz bilan uzatiladi. Pochta serverlari namunaga mos konfiguratsiya qilinishi shart.

Faraz qilamiz jo'natmalar linux.zukko.uz domeni uchun ikkita pochta serverlari smtp.zukko.uz va smtp.samtuit.uz -larda qabul qilinsin. U holda hududni tavsiflash fayliga quyidagi ikkita yozuvni qo'shish kerak bo'ladi:

```
@ IN MX 5 smtp.linear.zukko.uz .  
@ IN MX 10 smtp.samtuit.uz .
```

MX yozuvchi majburiy bo'lmagan yozuv hisoblanadi.

**A yozuvi.** A (address) turidagi yozuv kompyuter nomini IP adresga akslantirish uchun mo'ljallangan. Yozuvning formati quyidagicha:

```
[kompyuter nomi] [TTL] [IN] A IP adres
```

Faraz qilaylik, linux.zukko.uz domenida uchta kompyuterlar bor, bular: master.linear.zukko.uz, c1.linear.zukko.uz va c2.linear.zukko.uz. U holda hududni tavsiflovchi faylga quyidagi uchta yozuvni kiritish kerak bo'ladi:

```
master IN A 172.20.21.2  
c1 IN A 172.20.21.1  
c2 IN A 172.20.21.2
```

Kompyuterning IP adresi turli tarmoqlarda joylashgan bo'lishi mumkin.

Agar kompyuterda bir nechta tarmoq interfeyslari mavjud bo'lsa, u holda bitta kompyuter uchun ikkita yozuv yozish mumkin bo'ladi. Lekin, buni amalga oshirsak, bizni kutilmagan sovg'a kutadi. Masalan, c1.linear.zukko.uz kompyuterida ikkita tarmoq interfeysi 172.20.21.1 va 192.168.0.1 IP adreslari bilan mavjud deb olaylik.

Bundan hududni tavsiflovchi faylda quyidagi satrlar yozilgan bo'lsa:

```
c1 IN A 172.20.21.1  
IN A 192.168.0.1
```

U holda c1.linux.zukko.uz kompyuterining nomi birinchi so'rovda birinchi IP adresga beriladi. Ikkinchi so'rovda esa ikkinchi adresga beriladi. Uchinchi so'rovda yana birinchi adresga. To'rtinchi so'rovda yana ikkinchi adresga va h.k. Shuning uchun ham bitta kompyuterining DNS serveriga bitta IP adresni biriktirish tavsiya etiladi.

Ba'zida bu xususiyat yuklamani taqsimlash maqsadida qo'llanilishi mumkin. Masalan, WEB saytlarga xizmat ko'rsatuvchi ikkita WEB server mavjud bo'lsin. Agar www kompyuteridagi hududni tavsiflovchi faylda ikkita A yozuvni aniqlangan bo'lsa, u holda ushbu WEB serverlar o'rtasida so'rovlarni taqsimlash amalga oshiriladi. Haqiqatan ham, DNS server va klient keshiga tushgan ushbu server bir sutka davomida foydalanadigan yozuvlar WEB serverlardan faqat bittasiga borib tushadi.

**CNAME yozuvi.** CNAME (Canonical Name) – bitta kompyuterga bir nechta nomni biriktirish uchun qo'llaniladi.

CNAME yozuvining formati:

```
qo'shimcha nom [TTL] [IN] CNAME qat'iy nom
```

Bitta kompyuterga, jumladan, bitta IP adresga akslantiriladigan turli xil domenlarda bir nechta nom qo'yish mumkin. IP adresni teskari akslantirishda faqat bitta kompyuter nomiga o'tkaziladi. Bunday nom qat'iy nom deb ataladi. A yozuvi yordamida faqat qat'iy nom aniqlanadi. Barcha qo'shimcha nomlar faqat CNAME yozuvchi orqali amalga oshiriladi.

Masalan, master.linux.zukko.uz kompyuteriga qo'shimcha www.linux.zukko.uz va ftp.linux.zukko.uz nomlarini biriktrish masalasi qo'yilgan bo'lsin. U holda hududni tavsiflash fayliga quyidagi satrlar qo'shilishi kerak bo'ladi:

```
www IN CNAME master
```

```
ftp IN CNAME master
```



*CNAME yozuvining parametrlari kompyuter nomlari bo'lib, IP adres emasligini eslab qolish muhim*

**PTR yozuvi.** PTR (Pointer) turidagi yozuv IP adresni kompyuter nomiga teskari akslantirish uchun mo'ljallangan. Odatda bu yozuv

teskari akslantirish hududida qo'llaniladi.

Yozuvning formati:

**adres [TTL] [IN] PTR nom**

Bu yozuv haqida to'liq ma'lumot teskari akslantirish hududini yaratish misolida keltirib o'tiladi.

**O'qituvchining DNS serveriga linux.zukko.uz domenini qo'llab quvvatlashni qo'shish.**

Bu mavzuda BIND DNS serverida domenni qo'llab-quvvatlashni hosil qilishda bajarilishi zarur bo'lgan barcha holatlarni ko'rib chiqamiz.

### 5.9-§. named.conf faylining parametrlari

O'qituvchining DNS serverida linux.zukko.uz domenini qo'llashni hosil qilish kerak. Birinchi navbatda, named.conf konfiguratsiya faylida zone ko'rsatmasi yordamida domenni tavsiflash kerak bo'ladi. Buning uchun faylga quyidagi satrlar qo'shiladi:

```
zone "linux.zukko.uz " IN {  
    type master;  
    file "master.linear.zukko.uz " ;  
};
```

Misolda faqat zone ko'rsatmasidagi kerakli parametrlar keltirilgan:

type master – ko'rsatilgan hudud master hududi ekanligini aniqlaydi.

file "master.linear.zukko.uz " – domen kompyuteri va hudud parametrlari tavsiflanadigan fayl nomini aniqlaydi. BIND serveri bu faylni options ko'rsatmasida aniqlangan directoty parametri bilan berilgan katalogdan qidiradi. Fayl nomi ixtiyoriy bo'lishi mumkin, lekin DNS server bir nechta master va slave hududlarni qo'llab-quvvatlasa, master hududini tasviflovchi fayl nomi master so'zi bilan, slave hududini tavsiflovchi fayl nomi esa slave nomi bilan boshlanishi tavsiya qilinadi.

Master hududini tavsiflashda qo'llaniladigan boshqa parametrlar haqida keyingi mavzularda qarab chiqamiz.

## 5.10-§. Hududni tavsiflash fayli

linux.zukko.uz hududini tavsiflash uchun /var/named/master.linear.zukko.uz fayli hosil qilamiz.

Birinchi o‘rinda hududning barcha yozuvlari uchun jimlik holatida hayotiylik vaqtini aniqlaymiz:

```
$TTL 60
```

Real sharoitda bu qiymat bir sutkaga teng bo‘lishi kerak - 1D yoki shunga yaqin qiymat. Lekin hudud yozuvi sinfiga oid yozuvlarda doimiy o‘zgarishlar kiritilishi munosabati bilan TTL 60 sekundga tenglashtiriladi.

**SOA turidagi yozuvni hosil qilish.** Hududni tavsiflash faylida birinchi o‘rinda SOA turidagi yozuv bo‘lishi shart.

```
linux.zukko.uz IN SOA master.linear.zukko.uz  
. {  
    ilhom.linear.zukko.uz .  
    2006032300  
    18H  
    20M  
    2W  
    60 }
```

Agar nomni yozish xususiyatlari qo‘llasak va hududni tavsiflash faylidagi maxsus belgilardan foydalansak, birinchi satrni qisqartirish mumkin.

Birinchi, domen nomi o‘rniga @ belgisini qo‘yish mumkin.

Ikkinchi, misoldagi holatda master DNS serveri u qo‘llab-quvvatlaydigan domenda joylashgan, shuning uchun ham DNS serveri nomi o‘rniga qisqagina master so‘zini oxirida nuqtasiz yozish mumkin. Server domeni nomini o‘zimiz qo‘yishimiz mumkin.

Hududga mas’ul shaxsning elektron pochta adresi ham yuqoridagiga tegishli. Bu maydonga oxiriga nuqta qo‘ymasdan siz ilhom deb yozish mumkin. Server domen nomi avtomatik qo‘yiladi.

Natijada SOA yozuvi quyidagi ko‘rinishda bo‘ladi:

```
@ IN SOA master ilhom{  
    2006032300  
    18H  
    20M  
    2W  
    60 }
```

Oxirgi maydon - DNS server keshining hayotiy vaqti \$TTL direktivi uchun salbiy javob kiritilgan, bu bir sutkaga teng bo'lishi kerak. Lekin hudud yozuvi sinfiga oid yozuvlarda doimiy o'zgarishlar kiritilishi munosabati bilan TTL 60 sekundga tenglashtiriladi.

### 5.11-§. Domen uchun mas'ul DNS serverlarni aniqlash

Biz o'rganayotgan misollardagi linux.zukko.uz domeniga faqat bitta DNS server mas'ul hisoblanadi, ya'ni hududni tavsiflash faylida bitta NS turidagi yozuv kiritilgan.

```
@ IN NS master.linear.zukko.uz .
```

SOA yozuvidagidek NS yozuvida ham qiqartirishlar kiritishimiz mumkin. Birinchi maydon bo'sh qoldiriladi. Uning qiymati oldingi SOA yozuvidan olinadi. Hududga mas'ul DNS server nomini ham qisqartirish mumkin. Natijada yuqorida keltirilgan yozuv quyidagi ko'rinishga keladi:

```
IN NS master
```



*Birinchi bo'sh qoldirilgan maydon o'rniga probel belgisi yoki tabulyatsiyani qo'yish kerak.*

**Domen uchun jo'natmalarni qabul qiluvchi pochta serverini aniqlash.** linux.zukko.uz domeni uchun jo'natmalarni faqat bitta pochta serveri - master.linear.zukko.uz qabul qiladi. Shuning uchun ham hududni tavsiflash faylida MX turidagi yozuvni qo'shish kerak:

```
IN MX 5 master
```

MX turidagi yozuv birdaniga NS yozuvidan keyin keladi, shuning uchun ham birinchi maydonni ko'rsatmaslik ham mumkin, uning qiymati oldingi yozuvlardan olinadi.

**Domenga tegishli kompyuterlarni tavsiflash.** linux.zukko.uz domeniga tegishli barcha kompyuterlar A turidagi yozuv yordamida berilishi kerak.

```
master IN A 172.20.21.2
c1      IN A 172.20.21.1
c2      IN 1 172.20.21.2
```

va hakoza.

**Kompyuterlarning psemdonomlarini tavsiflash.** Barcha

pseudonimlar CNAME turidagi yozuv orqali beriladi.

```
www IN CNAME master
```

```
ftp IN CNAME master
```

```
pop IN CNAME master
```

Yuqoridagi yozuvlar kiritilgandan keyin hududni tavsiflovchi fayl quyidagi ko'rinishda bo'ladi:

```
$TTL 60
```

```
@ IN SOA master root (
```

```
2016032200
```

```
1BH
```

```
20M
```

```
2W
```

```
60 )
```

```
    IN NS master
```

```
    IN MX 5 master
```

```
master IN A 172.20.21.2
```

```
c1      IN A 172.20.21.1
```

```
c2      IN A 172.20.21.2
```

```
c3      IN A 172.20.21.3
```

```
c4      IN A 172.20.21.4
```

```
c5      IN A 172.20.21.5
```

```
c6      IN A 172.20.21.6
```

```
c7      IN A 172.20.21.7
```

```
c8      IN A 172.20.21.8
```

```
c9      IN A 172.20.21.9
```

```
c10     IN A 172.20.21.10
```

```
c11     IN A 172.20.21.11
```

```
c12     IN A 172.20.21.12
```

```
www IN CNAME master
```

```
ftp IN CNAME master
```

```
pop IN CNAME master
```

## 5.12-§. Sintaksis xatolarni tekshirish uchun utilitlar

BIND server tarkibida named.conf va hududni tavsiflash faylida bajarilgan o'zgartirishlardan keyingi sintaksis xatoliklarni tekshirish uchun ikkita utilit mavjud.

named-checkconf utiliti named.conf faylidagi sintaksis xatoliklarni tekshirish uchun mo'ljallangan. Bu dastur (utilit) yuklanishi uchun qo'shimcha parametrlarni talab etmaydi. Agar

tekshirilayotgan faylda xatoliklar bo‘lmasa, utilit ekranga hech qanday xabarni chiqarmaydi. Agar xatolik uchrasa, dastur ekranga xatolik topilgan satr nomerini chiqaradi.

Hududni tavsiflovchi fayldagi sintaksis xatolarni tekshirish uchun `named-checkzone` utilitidan foydalaniladi. Uning formati quyidagicha:

**`named-checkzone domen tavsif_fayli`**

Masalan, `linux.zukko.uz` hududini tavsiflovchi fayl sintaksisini tekshirish uchun quyidagi komandani kiritish kerak:

**`named-checkzone linux.zukko.uz \`  
**`/var/named/master.linux.zukko.uz`****

Agar faylda sintaksis xatolik topilsa, xatolik mavjud bo‘lgan satr nomeri ekranga chiqadi.



*Faqat konfiguratsiya fayllarining sintaksis xatoliklari tekshirilgandan keyingina DNS serverni ishga tushirish yoki qayta yuklash tavsiya etiladi.*

### 5.13-§. DNS serverni boshqarish

DNS serverni ishga tushirish, to‘xtatish va undan foydalanishning boshqa imkoniyatlarini turli dasturlar yordamida bajarish mumkin:

- BIND 9 serverida `rndc` dasturi yordamida.
- Initsializatsiya tizimining boshlang‘ich skriptlari yordamida.
- Signallar yordamida.

BIND 9 serverida boshqarishning maxsus `rndc` dasturi mavjud. Dasturni chaqirishda quyida keltirilgan komandalardan birini ko‘rsatish kerak.

Komanda	Tavsifi
<code>reload</code>	Barcha konfiguratsiya va hududni tavsiflash fayllarini o‘qiydi.
<code>reload zone [class [view]]</code>	Aniq hudud faylini o‘qiydi
<code>reconfig</code>	Konfiguratsiya faylini o‘qiydi va faqat yangi hudud faylini yuklaydi.
<code>stats</code>	Serverdagi maxsus <code>named.stats</code> fayliga statistikani yozadi
<code>dumpdb</code>	DNS serverning keshidagi ma’lumotlarni <code>named_dump.db</code> fayliga saqlaydi

stop	O'zgarishlarni saqlaydi va DNS serverni to'xtatadi.
halt	DNS serverni o'zgarishlarni saqlamasdan to'xtatadi.
trace	Otladka darajasini bir birlikka oshiradi. Agar otladka qo'shilgan vaqtda server oddiy rejimda ishlasa, barcha otladka ma'lumotlari joylashadigan named.run faylini hosil qiladi.
trace level	Otladkaning berilgan darajasini o'rnatadi.
notrace	Otladka rejimini o'chiradi.
flush	DNS serverning keshini tozalaydi.

### 5.14-§. DNS serverni yuklash

DNS serverni qo'shish uchun named-ning bajarilishini ta'minlash kerak. Buni komandalar satrida qo'lda yoki tizim initsializatsiyasining boshlang'ich skriptlari yordamida bajarish mumkin.

Named-ni ishga tushirish uchun unga quyidagi parametrlarni ko'rsatish kerak:

-u foydalanuvchi – ishga tushirish huquqi berilgan foydalanuvchini aniqlash. Parametr sifatida foydalanuvchining UID (foydalanuvchi identifikatsiya raqami) ko'rsatish yoki loginini kiritish mumkin.

-t katalog - chroot rejimida ishga tushirish.

Agar boshlang'ich skriptni qo'llamoqchi bo'lsak, u holda serverni yuklash quyidagi komanda yordamida amalga oshiriladi:

RedHat Linuxda: service named start

SuSE Linuxda: renamed start

Slackware Linuxda: /etc/rc.d/rc.bind start

**DNS server ishini to'xtatish.** DNS server ishini uchta usulda to'xtatish imkoni mavjud.

Signal yordamida:

killall named

Argument sifatida stop parametri kiritilgan boshlang'ich skript yordamida:

Rndc dasturi yordamida

rndc stop

**Konfiguratsiya fayli va hududni tavsiflash fayllarini qayta**



**yuklash.** Agar demonga<sup>7</sup> HUP signali yuborilsa, u barcha konfiguratsiya fayllari va hududni tavsiflovchi fayllarni qaytadan o'qiydi.

```
killall -HUP named
```

rncd dasturi DNS serverlarni ancha silliq boshqarishni ta'minlaydi.

```
rncd reload
```

Ushbu komandadan keyin barcha konfiguratsiya fayllari va hududni tavsiflash fayllari qayta o'qiladi.

Agar faqat aniq bir hududni qayta yuklash zarur bo'lsa, u holda reload zone komandasini qo'llash kerak.

```
rncd reload zone class.unix
```

### 5.15-§. DNS klienti sozlamalari

DNS klientining qaysi serverga so'rov yuborishni ko'rsatish uchun, /etc/resolv.conf faylida nameserver parametrlari yordamida DNS serverlarning IP adresini ko'rsatish kerak.

Bu faylda nameserver-ning uchtadan ko'p bo'lmagan parametrlarini aniqlash mumkin. Masalan,

```
nameserver 172.20.21.2  
nameserver 10.10.1.1
```

Shuningdek, ushbu faylda to'liq bo'lmagan nomni kiritish mumkin bo'lgan search parametrini aniqlash mumkin.

Search parametrlari sifatida probellar bilan ajratilgan domen nomlari qo'llaniladi. Masalan,

```
search ki.linux.zukko.uz linux.zukko.uz
```

Agar ping dasturiga parametr sifatida ki nomini bersak, uni DNS klientga aklantirish uchun berish vaqtida, klient uning nomida nuqta yo'qligini aniqlaydi va bu FQDN nomi emas deb topadi. Dastlab klient ki.linux.zukko.uz domen nomiga qo'yadi va uni akslantirish yuboradi. Agar javobda xatolik haqida xabarni olsa, ro'yxatga keyingi domenni qo'yadi.



1. named.conf faylida options ko'rsatmasi nima uchun

---

<sup>7</sup> Demon (daemon)- UNIX oilasidagi operatsion tizimlardagi kompyuter dasturlari bo'lib, bu dasturlarni foydalanuvchiga bog'liq bo'lmagan holda tizimning o'zi fon rejimida ishga tushiradi.

kerak?

2. Qanday ko'rsatma yordamida named.conf faylida BIND qo'llaydigan hududni tavsiflash mumkin?
3. Qanday faylda ";" belgisi izoh hisoblanadi?
4. Hududni tavsiflash faylida qanday turdagi yozuvlar albatta bo'lishi shart?
5. Qanday utilitlar yordamida named.conf va hududni tavsiflash fayllaridagi sintaksis xatoliklarni tekshirish mumkin?
6. DNS serverni boshqarish uchun qaysi dasturlar tavsiya etiladi?
7. Linuxda DNS klient konfiguratsiyasi uchun qanday fayl qo'llaniladi?

## 5.16-§. SLAVE hududini qo‘llash sozlamalari

BIND serverida slave hududini sozlash juda oddiy bo‘lib, named.conf faylida zone ko‘rsatmasini tavsiflash yetarli bo‘ladi.

```
zone "ki.linux.zukko.uz" IN {  
    type slave;  
    masters { 172.20.21.2; }  
    file "slave.ki.linux.zukko.uz ";  
};
```

Masters parametri majburiy hisoblanadi. U bizning DNS server hudud haqidagi ma‘lumotlar uchun murojaat qiladigan master serverning IP adresini aniqlaydi.

file parametri majburiy emas, lekin uni har e‘timolga qarshi tavsiflash kerak. Bu parametrda hudud haqidagi ma‘lumotlar saqlanadigan fayl aniqlanadi. Agar fayl aniqlanmagan bo‘lsa, u holda ma‘lumotlar faqat tezkor xotiraga saqlanadi va DNS serverning qayta yuklanishida master serveridan hudud haqidagi ma‘lumotlarni yangidan olishga to‘g‘ri keladi.

## 5.17-§. Domenni boshqarish huquqini tayinlash

Bu mavzuda domen huquqini olish uchun zarur bo‘ladigan ish faoliyati haqida so‘z yuritamiz.

Birinchi navbatda qism domen hosil qilinadigan domen egasiga murojaat qilish kerak. Masalan, ki.linux.zukko.uz domenidan foydalanmoqchi bo‘lsak, u holda linux.zukko.uz domenining egasidan ruxsat olishimiz va ki.linux.zukko.uz qism domenidan foydalanish shartlarini o‘rganib chiqishimiz kerak bo‘ladi.

Yuqori turuvchi domen egasining roziligisiz ham ki.linux.zukko.uz domenini hosil qilishimiz mumkin. Lekin, bunday hollarda bu domen faqatgina sizning DNS serveringizdan foydalanayotgan klientlarga ko‘rinadi xolos. Internetdan foydalanuvchilarning qolgan qismlari bu domendan umuman foydalana olmaydilar. Shuning uchun ham, yuqori turuvchi domen egasidan ruxsat olish majburiy hisoblanadi.

Agarda yuqori turuvchi domen egasi ki.linux.zukko.uz domenini boshqarish huquqiga rozilik bersa, u holda siz quyidagi ishlarni bajarishingiz kerak bo‘ladi:

- o‘zingizning DNS serveringizda domenni qo‘llab-quvvatlashni ishga tushirish;

- boshqa DNS serverlarning egalari bilan ularning serveri slave hududni qo‘llab-quvvatlash bo‘yicha kelishib olish. Buning uchun bir nechta slave DNS serverlar bo‘lishi tavsiya etiladi.

- domenga mas’ul bo‘lgan IP adres va DNS server nomini yuqori turuvchi domen egasiga berish.

Shundan keyin yuqori turuvchi domen egasi Sizga domenni boshqarish huquqini tayinlaydi. Yaratilgan domen boshqa Internet foydalanuvchilariga ko‘rinadi.

Domenni boshqarishga huquqni tayinlash yuqori turuvchining hududini tavsiflovchi fayliga qo‘shiladigan NS va A turidagi yozuvlar yordamida amalga oshiriladi.

Masalan, yuqori turuvchi domen egasiga quyidagi ma’lumotlar taqdim etilgan bo‘lsin:

- master DNS server - s2.ki.linux.zukko.uz , uning IP adresi – 172.20.21.2.

- slave DNS server ns.samtuit.uz , uning IP adresi – 195.158.1.1.  
linux.zukko.uz domenining hududni tavsiflash faylida quyidagi yozuvlar kiritiladi:

```
ki.linux.zukko.uz . IN NS
c2.ki.linux.zukko.uz .
c2.ki.linux.zukko.uz . IN A 172.20.21.2
ki.linux.zukko.uz . IN NS ns.samtuit.uz .
ns.samtuit.uz . IN A 195.158.1.1
```

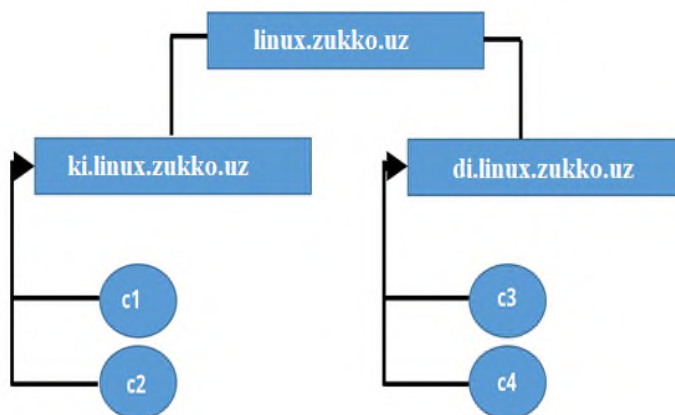
linux.zukko.uz mas’ul bo‘lgan hududni tavsiflash fayliga ushbu yozuvlar kiritilgandan keyin, ki.linux.zukko.uz hududiga mas’ul bo‘lgan DNS server IP ni berish mumkin.

### **5.18-§. Ichki domenlardan foydalanish**

Ba’zi hollarda ichki maqsadlarda foydalaniladigan domen hosil qilish zaruriyati tug‘iladi. Bunday domenlarga standart ierarxiyadan chiqqan holda odatda ixtiyoriy nom berish mumkin. Bunday domenlarga misol sifatida linux.zukko.uz domenini olish mumkin.

Bunday domenlarning asosiy kamchiligi – ierarxik zanjir bo‘yicha ildiz domendan standart vositalar yordamida kirish ruxsatini olib bo‘lmaydi. Agarda zukko.uz domeniga xizmat ko‘rsatuvchi DNS

serveriga master.linux.zukko.uz kompyuteri haqida so'rov berilsa xatolik haqidagi xabarni olamiz. Ya'ni zukko.uz domenining administratori bizga linux.zukko.uz domenini boshqarish huquqini tayinlamagan.



Agar tashkilot bo'yicha ichki foydalanish uchun domenlarga mas'ul bo'lgan bitta DNS server mavjud bo'lsa va barcha klientlar ishi faqat shu serverga sozlangan bo'lsa, hech qanday muammo tug'ilmaydi. Lekin, agar bizning kompyuter sinfimizdagiga o'xshash bir nechta domen va DNS serverlar qo'llanilayotgan bo'lsa, u holda adreslarni hal qilishda muammolar paydo bo'ladi.

O'z kompyuteringizda DNS server hosil qilingandan keyin, kompyuter nomini faqat o'zingizning domendan akslantirishingiz mumkin. Chunki, linux.zukko.uz domenidan yoki qo'shni juftlik domenidan kompyuter nomini akslantirishga harakat qilsangiz, DNS serveringiz ildiz serveriga murojaat qiladi va xato haqidagi xabarni oladi.

Yuqorida keltirilgan sxemadagi linux.zukko.uz hududiga mas'ul DNS serveri, boshqa DNS serverlarga ki.linux.zukko.uz va di.linux.zukko.uz domenlarini boshqarish huquqini beradi. Shuning uchun ham, unga c1.ki.linux.zukko.uz kompyuteri haqidagi so'rov bilan murojaat qilinsa, mos hududga mas'ul bo'lgan DNS server IP adresini biladi va unga qiziqtirgan kompyuter haqidagi so'rovlarni yuborish mumkin. Shuning uchun quyidagini amalga oshirish kerak, ya'ni, nufuzli bo'lmagan (zone ko'rsatmasi yordamida named.conf faylida hududi kiritilmagan) qo'shimcha DNS serverdan hudud haqidagi so'rov bilan master.linux.zukko.uz domenining DNS serveriga murojaat qiling. Buni bir nechta usullarda amalga oshirish mumkin.

**Forwarders parametri.** Birinchi usul kengroq tarqalgan. Bu

usul ierarxiyada yuqori turuvchi domenga mas'ul bo'lgan DNS server boshqarish huquqini bergan bitta darajadagi hududga mas'ul bir nechta DNS serverlar mavjud bo'lganda qo'llaniladi. Masalan, bitta kompyuter sinfidagi bir nechta DNS server misolida qarashimiz mumkin.

Bu DNS serverlar faqat bitta bo'limdagi klientlarni qo'llab quvvatlashga mo'ljallangan bo'lib, boshqa hech qanday funksiyalar mas'ulligini olmaydi.

Masalaning yechimi uchun `/etc/named.conf` konfiguratsiya faylidagi options ko'rsatmasiga quyidagi satrlarni qo'shish yetarli bo'ladi:

```
forwarders { 172.20.21.2; };
```

Bu parametr nufuzli bo'lmagan joriy DNS server uchun barcha hududlar haqidagi so'rovlarni ko'rsatilgan IR adresdagi serverga yuborilishini bildiradi. Bu holatda tizim bir me'yorda ishlashni boshlaydi va DNS server mustaqil ravishda ildiz serverga so'rovlarni yubormaydi.

**Forward turidagi hudud.** Faraz qilaylik bizda internetga chiqish uchun DNS server mavjud bo'lib, uning vazifasi tashkilotning ichki tarmog'idagi barcha DNS serverlarning so'rovlarini keshlashdan iborat bo'lsin. Hamda firewall-da Internetda joylashgan faqat DNS serverlarga so'rovlarni ushbu keshlovchi server orqali yuborishga ruxsat berilgan.

linux.zukko.uz hududiga mas'ul bo'lgan DNS-server Internetda joylashgan mashinalar nomlarini akslantirish bo'yicha so'rovlarga me'yorida xizmat ko'rsatishi uchun, unda ildiz domenning DNS-serverlari va boshqa Internet-domenlarga so'rovlarni yuborish imkoniyati mavjud bo'lishi kerak. Lekin firewall-da ularga to'g'ridan to'g'ri kirish yopib qo'yilgan.

Barchasi me'yorida ishlashi uchun ushbu DNS serverdagi konfiguratsiya fayli `named.conf`-da tashkilotning keshlovchi DNS serverini ko'rsatuvchi `forwarders` satrini kiritish kerak bo'ladi:

```
forwarders { 10.10.1.1; };
```

Shundan keyin tizim me'yorida ishlashi kerak. Lekin, afsuski, bunday bo'lmaydi. Madomiki, biz linux.zukko.uz hududiga mas'ul bo'lgan DNS serverga `forwarders` parametrini qayd qilganimiz zone ko'rsatmasi yordamida tavsiflangan u nufuzli bo'lgan hudud haqidagi

so'rovlarga javob berishini anglatadi. Ya'ni `ki.linux.zukko.uz` , `di.linux.zukko.uz` va boshqa domenlaridagi zone ko'rsatmasi yordamida ushbu DNS server ular haqida so'rov yuborishini aniqlovchi `forwarders` parametri kiritilmagan. Shuning uchun ham keshlovchi DNS server bunday so'rovlarni ildiz DNS serveriga yo'naltiradi va xatolik haqidagi xabarni oladi.

Bu muammoni yechish uchun `linux.zukko.uz` domeniga mas'ul DNS serverdagi `named.conf` faylida zone ko'rsatmasi yordamida `ki.linux.zukko.uz` , `di.linux.zukko.uz` va boshqa domenlar hududini tavsiflash kerak bo'ladi. Bizning DNS server bu hudud uchun master hisoblanmaydi, shuning uchun ham master turini qo'llash mumkin emas. Slave hududni hosil qilish mumkin, lekin bu hamma vaqt ham to'g'ri yechim bo'lmaydi. Bizning misolda hududni tavsiflashda `forward` turini tanlash to'g'ri yechim hisoblanadi.

Masalan, `ki.linux.zukko.uz` hududi uchun quyidagi satrlarni qo'shish kerak:

```
zone "ki.linux.zukko.uz " IN {
    type forward;
    forwarders { 172.20.21.2; };
};
```

Endi, agar bizning DNS serverga kompyuterlar nomini akslantirish bo'yicha so'rov `ki.linux.zukko.uz` domenidan kelsa, u so'rovni zone ko'rsatmasida `forwarders` parametri yordamida tavsiflangan DNS serverga yo'naltiradi.



1. Qanday parametr yordamida nufuzli bo'lmagan DNS server hududi haqidagi barcha so'rovlarni provayderning DNS serveriga yo'naltiradi?
2. `forward` turidagi hudud nima uchun qo'llaniladi?

## 5.19-§. DNS serverni testlash va tuzatish (otladka qilish)

DNS serverni testlash va tuzatish uchun bir nechta usullardan foydalanish mumkin:

- Log fayl mazmunini ko'rish.
- Serverni tuzatish (otladka) rejimida yuklash.
- DNS serverga so'rovlarni yuboruvchi dasturlardan foydalanish. named dasturini ishga tushirish jarayonida log fayliga turli xil ma'lumotlar joylashtiriladi:

- dastur qabul qiladigan so'rovlar va komandalar interfeyslari.
- qaysi hudud yuklangan yoki qayta yuklanganligi.

Afsuski, log fayliga kelib tushadigan ma'lumotlar mukammal ma'lumot hisoblanmaydi va ba'zida DNS serverning ishida aniqlangan xatoliklar uchun ma'lumotlar yetarli bo'lmaydi.

BIND log fayliga qanday hajmdagi ma'lumotlar tushishi zarurligini tanlashga yo'l qo'yadi. Buning uchun logging ko'rsatmasi qo'llaniladi. Lekin uni qo'llash sozlash uchun sarflangan vaqtni oqlamaydi. Bundan tashqari serverning tuzatish rejimi ba'zida kerak bo'ladi. Shuning uchun ham otladka qilish uchun komandalar satri orqali rndc dasturini yoki DNS serverga so'rovlar yuboruvchi dasturlarni qo'llash kerak bo'ladi.

**DNS serverning tuzatish darajalarini qo'llash.** DNS serverni otladka qilish darajalari 0 dan 11 gacha belgilanadi. Qancha katta daraja ko'rsatilsa, dastur shuncha mukammal ma'lumotlarni chiqarish imkoniyatiga ega bo'lamiz.

0-daraja otladka qilish rejimini o'chiradi. 1- va 2-darajalari konfiguratsiya fayli va hududni tavsiflash fayllarini otladka qilish uchun qo'llaniladi. Qolgan barcha darajalarni BIND-ni ishlab chiquvchilar qo'llaydi.

Otladka rejimini ishga tushirishning eng sodda yo'li - named dasturini -d parametri bilan yuklash.

**named -d2**

Dasturni ishga tushirgandan keyin barcha otladka ma'lumotlari /var/named/named.run fayliga joylashtiriladi.

Otladka rejimini yoqish uchun rndc utilitidan ham foydalanish mumkin.

**rndc trace 2**

Bu holatda named.run fayli hosil bo'ladi.

Otladka rejimini o'chirish uchun ham rndc utilitidan foydalanish



mumkin:

### **rndc notrace**



*Har qanday holatda ham otladka rejimini ishga tushirishdan oldin named-checkconf va named-checkzone dasturlari yordamida sitaksis xatoliklarni tekshirish va xatolarni tuzatish tavsiya etiladi.*

**Nslookup, dig va host dasturlaridan foydalanish.** DNS serverga so'rovlar yuborish uchun quyidagi dasturlardan foydalanish mumkin:

- nslookup.

- dig – dasturi aynan nslookup dasturining barcha funksiyalarini bajaradi, lekin bu dasturning jimlik holatidagi parametrlari ko'proq samarali hisoblanadi. Bu dastur ko'proq ma'lumot beradi va foydalanish uchun juda qulay.

- host – dasturi qisqa ma'lumotlar beradi.

**Nslookup dasturi.** Dastur ikkita rejimda ishlashi mumkin:

- barcha parametrlar komandalar satrida beriladigan rejim.

- interaktiv rejim.

Quyidagi jadvalda interaktiv rejimda ishlatish mumkin bo'lgan ba'zi komandalar berilgan.

<b>Ko'rsatma</b>	<b>Tavsifi</b>
exit	Dasturdan chiqish
server	Jimlik holatidagi so'rov yuboriladigan serverni tanlash. Jimlik holatida DNS server /etc/resolv.conf faylidagi tavsifdan foydalanadi.
set type=tur	So'rov uchun yozuv turini o'rnatish

**Dig dasturi.** Dig dasturi interaktiv ish rejimiga ega emas, barcha javoblarni hududni tavsiflash fayli formatida standart chiqishda taqdim etadi.

```
dig [paramerlar] nom [yozuv turi]
[@DNS_server]
```

Dig dasturiga ma'lumot zarur bo'lgan kompyuter nomi yoki domen nomini ko'rsatish kerak. Shuningdek, qiziqtirgan yozuv turini ham aniqlash mumkin.

Dastur jimlik holatida DNS serverga /etc/resolv.conf faylida

aniqlangan so‘rovni yuboradi. Agar qandaydir boshqa serverga yuborish zarur bo‘lsa, u holda shu serverning IP adresini @ belgisidan keyin ko‘rsatish kerak bo‘ladi.

Agarda IP-dan kompyuter nomiga teskari tartibda akslantirish haqidagi ma‘lumotni olishni xohlasak, u holda dasturni chaqirish parametriga –x ni qo‘yish kerak bo‘ladi.

**Host dasturi.** Bu dastur qisqa ma‘lumot beradi. Bu dasturni chaqirishda zarur bo‘lgan kompyuter nomini kiritish kerak bo‘ladi.

Yuqorida sanab o‘tilgan dasturlar ishlashi natijasining nimasiga e‘tibor berish kerak? Birinchi navbatda, DNS server qanday javob qaytarishi muhim. Odatda bunday usullar bilan oxirida ildiz domen ko‘rsatilmaydigan kompyuter va domenlar nomi aniqlanadi.

Agar dastur DNS serverdan javob olmasa, demak /etc/named.conf faylida uni noto‘g‘ri aniqlaganmiz yoki qandaydir sabarlar bilan serverga kirish yopiq.



1. DNS serverdagi xatoliklarni tekshirish uchun administratorga otladka qilishning qaysi darajalari tavsiya etiladi?
2. Otladka rejimi qo‘shilganda otladka haqidagi ma‘lumotlar qaysi faylga tushadi?
3. Otladka rejimi qanday qo‘shiladi (ishga tushiriladi)?

## 5.20-§. Teskari akslantirish hududi

Oldingi mavzularda to‘g‘ri akslantirishga oid barcha misollarda kompyuter nomini IP adresga akslantirish ko‘rib chiqildi. DNS server teskari akslantirish – ya‘ni IP adresni kompyuter nomiga o‘tkazish amalini ham bajara oladi.

Bu funksiyalarni qanday amalga oshirilishini tushinish uchun, in-addr.arpa domenida qism domenlar qanday shakllanishi prinsiplarini tushinib olish kerak bo‘ladi. Bu maxsus domen bo‘lib, teskari akslantirishga mo‘ljallangan.

Agar master.linux.zukko.uz kompyuterining nomini olib qaraydigan bo‘lsak, u ikkita qismdan tashkil topgan: master - nomi, linux.zukko.uz - domen.

Bu kompyuterning IP adresi (172.20.21.2-ga teng, netmask 255.255.255.0-ga teng bo‘lsin) ham ikkita qismdan iborat: 172.20.21. – tarmoq adresi, 2 – tarmoqdagi kompyuter adresi. Agar ushbu IP

adres teskarisiga 2.21.20.172 kabi yozilsa ham, 2 – kompyuter nomi, 21.20.172 – domen nomi ekanligini anglash mumkin.

Maxsus domen in-addr.arpa ni qo'llash hisobidan FQDN quyidagi ko'rinishga keladi:

**2.21.20.172.in-addr.arpa**

Xuddi shunday, 172.20.21/24 tarmog'ida IP adreslarni akslantirish uchun named.conf faylida 21.20.172.in-addr.arpa hududini aniqlash kerak bo'ladi. U master yoki slave bo'lishi mumkin. Hududni tavsiflash faylida SOA va NS yozuvlarini aniqlangan bo'lishi shart. IP adresni akslantirish uchun PTR turidagi yozuvdan foydalaniladi.

**ip\_nomi IN PTR o'zgarmas\_nom**

172.20.21.2 IP adresini akslantirish uchun hududni tavsiflash fayliga quyidagi yozuvni qo'shish kerak

**2 IN PTR master.linux.zukko.uz**

Teskari akslantirish hududlari IP adres hududining egasi tomonidan shakllantirilishi shart. Agar provayder sizga IP adreslar hududini to'liq ajratmagan bo'lsa, u holda unga teskari akslantirish hududiga kiritish uchun kompyuterlar nomini berishingiz kerak bo'ladi.



1. IP adresni kompyuter nomiga akslantirish uchun qanday turdagi yozuv qo'llaniladi?

2. Teskari akslantirish hududi kim tomonidan yaratiladi?

## 5.21-§. Keshlovchi server

DNS serverlar ko'pincha hech qanday hududni qo'llab quvvatlashsiz keshlovchi server sifatida qo'llaniladi. Bunday serverlar yetarli darajada katta sondagi trafikni tejash imkonini beradi.

BIND keshlovchi DNS server bo'lsa, serverning konfiguratsiya faylida quyidagi uchta hudud tavsiflangan bo'lishi shart:

- localhost domeni uchun to'g'ri akslantirish hududi.
- 0.0.127.in-addr.arpa teskari akslantirish hududi.
- ildiz domenni tavsiflovchi hudud.

Keshlovchi DNS serverning me'yorda ishlashi uchun, unga qandaydir usulda ildiz domenga mas'ul bo'lgan DNS serverlar IP adreslarini ko'rsatish zarur. Chunki, lokal keshda kompyuter haqidagi

ma'lumotlar bo'lmasa, dastlab ildiz domenga murojaat qilinadi va ierarxiya bo'yicha kerakli hudud uchun mas'ul bo'lgan DNS serverlargacha yetib boradi.

Ildiz DNS serverlarning IP adreslari maxsus hududdagi hint turi yordamida tavsiflanadi. Uning asosiy vazifasi ushbu IP adresni bizning serverga ildiz domenga murojaat qilishi mumkingini eslatib turishdan iborat.

```
zone "." IN {  
    type hint;  
    file "caching-examples/named.ca";  
};
```

Hint turidagi hududni aniqlashda majburiy bo'lgan parametr eslatma saqlanayotgan faylni ko'rsatuvchi file parametri hisoblanadi.

Bu faylda ildiz domenga mas'ul DNS serverni aniqlovchi NS va A turidagi yozuvlar juftligi saqlanadi.

```
.360000 IN NS A.ROTT-SERVERS.NET.  
A.ROOT-SERVERS.NET. 360000 A 198.41.0.4
```

Agar tasodifan eslatma fayli o'chirib yuborilsa, uni boshqa DNS serverdan dig utiliti yordamida olish mumkin:

```
dig . ns @IP_DNS > file
```

yoki ftp serveri yordamida ftp.internic.net/domain/named.cache.



1. DNS serverga ildiz hududga mas'ul bo'lgan serverlar adresini ko'rsatish uchun qanday hudud turi qo'llaniladi?

2. Ildiz hududga mas'ul bo'lgan DNS serverlar adreslari va nomlari haqidagi eslatma sifatida qo'llaniladigan faylni qanday qilib olish mumkin?

## 5.22-§. Xavfsizlik masalalari

BIND DNS serverni ishlatish jarayonida qanday potentsialda xavf (tahdid)lar paydo bo'lishi mumkinligini tasavvur qila olishimiz kerak.

Bu xavflarni ikkita qismga ajratish mumkin:

- DNS tizimidagi muammolar.
- named dasturidagi muammolar.

**DNS serverga kirish ruxsatidagi cheklovlar.** Birinchi ko'rinishdagi muammo DNS serverning ochiqligi bilan bog'liq. Yovuz niyatli kishilar (buzg'unchilar)da tahdidlarni tashkil etish uchun DNS-dan foydalanishning bir nechta imkoniyatlari bor:

- tashkilot tarmog'i haqida ma'lumotlarni olish.
- DNS serverdagi ma'lumotlarni o'zgartirish (almashtirish).

**Ma'lumotlarga kirishni cheklash.** Buzg'unchilar, sizning tarmog'ingizga tahdid qilishi uchun tarmoq topologiyasi va tarmoqdagi qaysi kompyuter qanday funksiyalarni bajarishini bilishi-tushunishi kerak. Bu parametrlarni A turidagi yozuv yordamida hisoblab chiqish mumkin.

Bunday turdagi xavflarga qarshi himoya BIND serveri ichiga o'rnatilgan - nslookup va dig kabi standart dasturlardan foydalanib, DNS serveridan A turidagi barcha yozuvlarni olish mumkin emas.

Lekin Siz toq nomerli kompyuterlarda slave hududni hosil qilgan vaqtingizda, DNS server hudud haqidagi barcha ma'lumotlarni - hatto, u hudud NS yozuvi yordamida aniqlangan nufuzli serverlar ro'yxatida bo'lmasa ham oladi.

Ushbu muammoni bartaraf etish uchun hududiy jo'natmalarni cheklash imkoniyatini beruvchi allow-transfer parametrini named.conf fayliga qo'shish kerak. Unga parametr sifatida slave serverlarning IP adreslarini ko'rsatiladi:

```
allow-transfer { 195.158.1.1; 172.20.20.20;  
};
```

yoki slave serverlar qo'llanilmayotgan bo'lsa, none kalit so'zi ishlatiladi.

Parametrni options ko'rsatmasida ham aniqlash mumkin. Bunday holda Sizning DNS serveringizning barcha hududlari uchun umumiy bo'ladi. Yoki har bir hudud uchun alohida aniqlash zarur.

Agar DNS serveri faqatgina ichki ehtiyojlar uchun foydalanilsa yoki faqatgina ba'zi hududlar Internetdan foydalanish uchun

mo'ljallangan bo'lsa, u holda BIND foydalanuvchilarga DNS serverdan qaysi tarmoq yoki IP adreslar orqali kirishga ruxsat cheklovlarini taqdim etadi.

Bunday cheklovlar uchun allow-query parametridan foydalaniladi.

```
allow-query { 172.20.20/24; 195.158.1.1; };
```

Bu parametrni options ko'rsatmasida aniqlash mumkin, u holda DNS serverga so'rovlar yuborishni umuman cheklaydi. Agar unda ichki hudud aniqlangan bo'lsa, u holda cheklov faqat shu hududga tegishli bo'ladi.

### **5.23-§. DNS serverlarda ma'lumot almashish**

Bu muammo DHCP serveri hududidagi ma'lumotlarni avtomatik yangilash imkoniyatini yaratish uchun kengaytirilgan DNS protokollarini qo'llab-quvvatlashda paydo bo'lishi mumkin.

Odatda hududdagi barcha kompyuterlarda ma'lumotlarni o'zgartirish mumkin. Hudud mazmunini o'zgartirish mumkin bo'lgan kompyuterlar ro'yxatini aniqlash uchun allow-update parametri qo'llaniladi.

```
allow-update { 195.158.1.1; };
```

Hududdagi ma'lumotlarni o'zgartirish imkoniyatidan foydalanmoqchi bo'lmasangiz, bu imkoniyatni none kalit so'zi yordamida taqiqlab qo'yish mumkin.

```
allow-update { none; };
```

**TSIG texnologiyasi qo'llash.** Oldingi mavzularda tavsiflangan IP adreslar bo'yicha nazorat qilish parametrlari haqida fikr yuritgan edik, bu nazorat usuli himoya qilishning ishonchli usuli emasligini tushunib olishimiz kerak. Agar tarmoqda hududdagi ma'lumotlarga o'zgartirish kiritishga ruxsat beruvchi DHCP server mavjud bo'lsa, buzg'unchilar uning IP adresini qo'llab, undan foydalanishlari mumkin. Xuddi shunday, slave server hududiga jo'natma yuborish vaqtida ma'lumotlarni o'zgartirish ehtimoli ham mavjud.

Bu masalalaning yechimini olish uchun TSIG texnologiyasidan foydalanish mumkin. Bu texnologiya ma'lumotlarni yuborishda uzatuvchi-qabul qiluvchi juftligini autentifikatsiya qilish va yuborilgan ma'lumotda o'zgarishlar bo'lganligini tekshirishda qo'l keladi.

TSIG texnologiyasida simmetrik shifrlash metodi qabul qilinadi. O‘zaro ma’lumot uzatayotgan har bir serverlar juftligi o‘zining kalitini hosil qilishi shart.



*TSIG texnologiyasini serverlar o‘rtasida ma’lumotlarni uzatishda qo‘llash mumkin. DNS klientlari esa bu texnologiyadan foydalana olmaydilar*

TSIG texnologiyasida qo‘llanilayotgan tranzaksiya imzolari, ma’lumotlar qabul qilinayotgan joyda tekshiriladi, ya’ni ular server keshida saqlanib qolmaydi.

Kalitlarni generatsiya qilish (hosil qilish)da dnssec- keygen dasturidan foydalaniladi. Bu dastur BIND 9 da qo‘yilgan. Faraz qilamiz, bizda ikkita c1.ki.linux.zukko.uz va c2.ki.linux.zukko.uz serverlari qo‘llaniladi, bunday hollarda kalitlarni generatsiya qilish uchun quyidagi komandani bajarish kerak bo‘ladi:

```
dnssec-keygen -a HMAC-MD5 -b 128 -n HOST  
c1.ki.linux.zukko.uz -c2.ki.linux.zukko.uz
```

Natijada ikkita fayl yaratiladi:

Kc1.ki.linux.zukko.uz -c2.ki.linux.zukko.uz .+157+29237.private  
va

Kc1.ki.linux.zukko.uz -c2.ki.linux.zukko.uz .+157+29237.key.

Bu yerda 157- MD5 algoritmining kodi, 29237-serverlarning bitta juftligida bir nechta kalitlar bor bo‘lgan holatda kalit identifikatori sifatida qo‘llaniladigan tasodifiy son.

Kc1.ki.linux.zukko.uz -c2.ki.linux.zukko.uz  
.+157+29237.private faylining mazmuni quyidagi ko‘rinishda bo‘lishi mumkin:

```
Private-key-format: v1.2  
Algorithm: 157 (HMAC_MD5)  
Key: y3AnkehA+ji36hTFTDx3MA==
```

Kc1.ki.linux.zukko.uz -c2.ki.linux.zukko.uz .+157+29237.key fayli qo‘llanilmadi va dnssec-keygen dasturi yordamida generatsiya qilinganligining sababi, ochuvchi va yopuvchi kalitlar juftligini generatsiya qilish uchun qabul qilingan.

Kalit ikkala server uchun ham bir xil bo‘lishi kerak. Bu kalitlarga kirish ruxsati uchun 600 kodli huquqni o‘rnating va named dasturida yuklash huquqi bilan foydalanuvchiga yuboring.



*Hech qachon faylni ochiq holatda elektron pochta yoki ftp orqali yubormang, bu xavfsiz emas. Uni disketada uzating yoki yuborishdan oldin gpg dasturi yordamida shifrlab oling.*

BIND-da kalitni aniqlash uchun /var/named katologida c1-c2.key faylini hosil qilib oling. Bu faylning mazmuni quyidagicha bo'ladi:

```
key "c1-c2" {  
    algorithm hmack-md5;  
    secret "y3AukehA+jI36hTFTDx3MA==" ;  
};
```

Bu faylga kirish ruxsati faqat named dasturini ishga tushirish huquqiga ega bo'lgan foydalanuvchida bo'lishi shart.

Bu faylni asosiy konfiguratsiya fayli named.conf ga qo'shish include ko'rsatmasi yordamida amalga oshiriladi:

```
include "c1-c2.key" ;
```

Kalit aniqlangandan keyin, named dasturini boshqa serverni aniqlash (identifikatsiya qilish)ga sozlash kerak:

```
server c2.ki.linux.zukko.uz {  
    keys { c1-c2; };  
};
```

Xuddi shunday bizning DNS serverimiz c2.ki.linux.zukko.uz serverida autentifikatsiyani aniqlashi kerak.

Barcha allow-update, allow-query va allow-transfer direktivalari ushbu kalitga murojaat qilishi kerak. Masalan,

```
allow-transfer { keys c1-c2; };
```

## **5.24-§. DNS serverni oddiy foydalanuvchi huquqida ishga tushirish**

Agar DNS server komandalar satrida root foydalanuvchisida ishga tushirilgan bo'lsa, server super foydalanuvchi huquqi bilan ishlaydi. Bu noto'g'ri. Serverning ishlashi uchun oddiy foydalanuvchi huquqi ham yetarli bo'ladi.

Serverni boshqa foydalanuvchi huquqi bilan ishga tushirish juda



oson bajariladi. Buning uchun –u parametri va loginni yoki foydalanuvchi UID-ini ko‘rsatish yetarli.

Agar Slackware Linux-da yuklovchi skript /etc/rc.d/rc.bind ga qarasaq, server super foydalanuvchi huquqida ishga tushirilishi ma’lum bo‘ladi. Buni tuzatish uchun dasturni ishga tushirgandan keyin darhol –u parametrini kiritish yetarli bo‘ladi:

**named –u daemon**

Lekin foydalanuvchini o‘zgartirgandan keyin uni /var/named va /var/run/named katologlarining barcha kontentlariga kiritib chiqishni unutmaslik kerak.

**DNS serverni chroot-da yuklash.** Afsuski, serverlarni BIND orqali buzish tez-tez uchrab turadi. Buzish imkoniyatlarini kamaytirish uchun BIND-ni doimiy ravishda yangilab turish zarur. Hech qachon uni root foydalanuvchisi huquqi bilan ishga tushirmaslik kerak. Imkoni boricha uni chroot dasturi yordamida ishga tushirish tavsiya etiladi.

Chroot dasturi ildiz katologni boshqasi bilan almashtiradi. Shundan so‘ng chroot yordamida ishga tushirilgan dastur fayl tizimida ko‘rsatilgan katologdan tashqarida “ko‘rinmaydi”.

Lekin, u boshqa katologlarni ko‘rmaganligi uchun, ushbu dastur ishlashiga zarur bo‘ladigan barcha fayllarni dastur uchun aniqlangan ildiz katologga ko‘chirib o‘tkazishga to‘g‘ri keladi. Bundan tashqari, ushbu katologda ildiz katolog kabi ishlashi uchun barcha katologlar daraxtini hosil qilish kerak.

Named dasturini ishga tushirishda –t paramterini ildiz katolog sifatida qo‘llaniladigan katolog nomini ko‘rsatish mumkin. Ya’ni, DNS server chroot dasturini oshkormas holda chaqirish bilan ishga tushiriladi.

Agar chroot-da named-ni ishga tushirmoqchi bo‘lsak, quyidagilarni amalga oshirishimiz kerak:

1. Serverni –t /var/named parametri bilan ishga tushirish.
2. /var/named katologida barcha zurur bo‘ladigan fayl va katologalarni hosil qilish<sup>8</sup>.
3. named.conf faylini /var/named/etc katologiga ko‘chirib o‘tkazish va unga /etc/named.conf belgili murojaat hosil qilish.
4. Hududni tavsiflovchi barcha fayllarni /var/named/var/named katologiga ko‘chirib o‘tkazish.

---

<sup>8</sup> Қандай файлларни ҳосил қилиш Chroot-BIND-HOWTO-да тўлиқ ёзилган.



1. Qanday parametrlar zone ko'rsatmasida hududiy jo'natmalar imkoniyatini cheklash uchun albatta qo'llanilishi shart?
2. Qanday parametrlar hudud tavsifini o'zgartirishni taqiqlash uchun zone ko'rsatmasida qo'llanilishi kerak?
3. TSIG texnologiyasini klient va DNS o'rtasidagi muloqotda qo'llash mumkinmi?
4. Qanday sabablarga ko'ra DNS serverni chroot yordamida ishga tushirish kerak?

## 6-BOB. ELEKTRON POCHTA

Linux serverida ishlovchi muhim tizimlardan biri Elektron pochta tizimi hisoblanadi. Linux administratori elektron pochta qanday sozlanishini tushinishi va uni sozlashni bilishi shart.

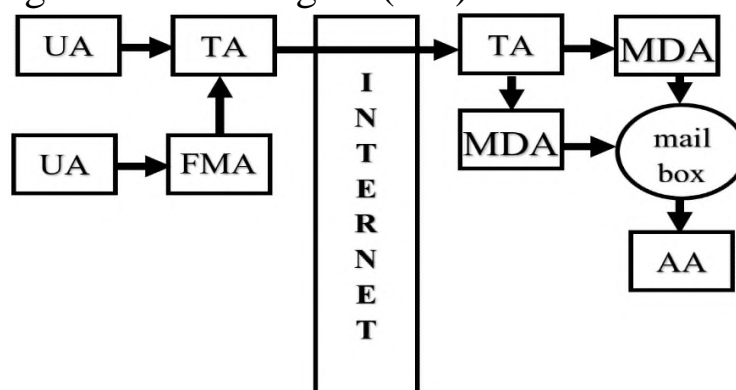
Ushbu bobda elektron pochta tizimi vositalari bilan tanishamiz va transport agenti sendmail ni sozlashni o'rganamiz.

### 6.1-§. Elektron pochta komponentlari

Elektron pochta tizimi bir nechta komponentlardan tashkil topgan. Ular bir yoki bir nechta dasturlar ko'rinishida bajarilishi mumkin.

Quyidagi komponentlarni keltirish mumkin:

- Foydalanuvchi agenti - User agent (UA).
- Transport agenti - Transport agent (TA).
- Pochtani uzatish agenti - Filing mail agent (FMA).
- Pochtani yetkazib berish agenti - Mail delivery agent (MDA).
- Kirish agenti - Access agent (AA).



**Foydalanuvchi agenti.** Foydalanuvchi agenti yordamida foydalanuvchilar xatni hosil qiladi va yuboradi. Agent xat tanasini standart asosida shakllantirishi va uni jo'natish uchun transport agenti yoki uzatish agentiga berishi kerak.

Foydalanuvchi agenti sifatida quyidagi dasturlar qo'llaniladi: The Bat, Outlook va Outlook Express. Agarda Linux tizimini e'tiborga olsak, u holda: Evolution, KMail, pine va boshqalarni keltirish mumkin.

Xatlarni transport agenti va uzatish agentiga taqdim etishda SMTP protokoli qo'llaniladi.

**Transport agenti.** Transport agenti ikkita asosiy vazifani

bajaradi:

- foydalanuvchi agentidan jo‘natmalarni qabul qilish va uni boshqa transport agentlariga yuborish;

- Boshqa transport agentlaridan jo‘natmalarni qabul qilish.

Foydalanuvchidan jo‘natmani qabul qilishda bu agent yuoriladigan manzilning to‘g‘riligini, jo‘natmani yetkazish imkoniyatlarini tekshirishi va ko‘rsatilgan manzil bo‘yicha jo‘natmani yetkazish.

Boshqa tomondan olib qaraganda, foydalanuvchining pochta qutisi aynan shu kompyuterda bo‘lsa, transport agenti ushbu xat (jo‘natma)ning shu kompyuter (domen)ga mo‘ljallanganligini tekshiradi. Tekshirishlardan keyin xatni qabul qiladi va uni jo‘natmani yetkazib berish agentiga uzatadi.

UNIX dunyosida transport agenti funksiyalarini tatbiq qiluvchi ko‘p sondagi dasturlar mavjud. Ularning ichida keng ommalashganlari, bepul tarqatiladigan sendmail, postfix, exim va qmail kabilarni ajratib ko‘rsatish mumkin. Har bir dasturning o‘ziga xos yutuq va kamchiliklari mavjud, eng ko‘p tarqalgani sendmail hisoblanadi – bu agent eng birinchi transport agentlaridan biridir.

**Pochtani uzatish agenti.** Pochtani uzatish agenti – bu transport agenti ish rejimining boshqacha ko‘rinishlaridan biridir. Uzatish agentlari yuqori yuklanishli trafikka ega bo‘lgan pochta tugunlarida qo‘llaniladi. Uning vazifasi transport agentining asosiy ishlarini yengillashtirish hisoblanadi.

Uzatish agentining funksiyalari:

- tugun nomlari to‘liq aniqlanganligini tekshirish.

- noto‘g‘ri ishlayotgan foydalanuvchi agentidan olingan xabar sarlavhalarini modifikatsiyalash.

- Xatni transport agentiga yuborishdan oldin barcha xatoliklarni tekshirish.

Uzatish agenti 587 portidagi so‘rovlarni tekshiradi, shuning uchun ham barcha foydalanuvchi agentlari jo‘natmalarni aynan shu portga uzatishga moslashtirilgan holda konfiguratsiya qilinishi kerak.

Uzatish agentini ishining barcha xususiyatlari RFC 2476-da tavsiflangan.

**Pochtani yetkazib berish agenti.** Transport agenti jo‘natmani olgandan keyin, jo‘natmalarni foydalanuvchi qutisiga yetkazmaydi. U jo‘natmani yetkazib berish agentiga yuboradi, yetkazib berish agentining vazifasi jo‘natmani foydalanuvchi qutisiga uzatadi.

Yetkazib berish agenti sifatida jo‘natmalarni yig‘uvchi oddiy dasturlardan foydalaniladi. Ammo ancha murakkab dasturlar ham mavjud. Jo‘natmalarni yetkazib berishda ularni filtrlashi mumkin bo‘lgan dasturlar mavjud, masalan, rgosmail.

Pochtani yetkazib berish agenti nima uchun mo‘ljallanganligi RFC2476-da tavsiflangan.

**Kirish agenti.** Kirish agenti foydalanuvchi o‘zining pochta kirishga ruxsatni olishda qo‘llaniladi. bular ham dastur ko‘rinishida bajarilib, pochta qutisiga pop yoki imap protokollari bo‘yicha kirishni tashkil qilib beradi.

Agar foydalanuvchi uning pochta qutisi saqlanayotgan kompyuterda lokal ishlayotgan bo‘lsa, kirish agentsiz pochta qutisiga kirishni amaga oshirishi mumkin. Xuddi shunday prinsip asosida mail va pine dasturlari ishlaydi.

## 6.2-§. SMTP protokoli

Elektron pochta tizimida o‘zaro aloqaning asosiy protokoli sifatida SMTP (Simple Mail Transfer Protocol) va ESMTP (Extended SMTP) lar qo‘llaniladi. Bular RFC2821, 1869, 1870, 1891 va 1985 larda tavsiflangan.

SMTP protokoli oddiy aloqa protokoli sifatida yaratilgan bo‘lib, foydalanuvchi to‘g‘ridan-to‘g‘ri transport agenti yordamida muloqot qilishi mumkin. Albatta, foydalanuvchi transport agenti bilan ishlamaydi. Ishni yengillashtirish uchun foydalanuvchi agenti qo‘llaniladi. Lekin, foydalanuvchi agentlari transport agentlari bilan aloqani SMTP yoki ESMTP protokollari orqali amalga oshiradi.

Misol sifatida foydalanuvchi qanday qilib telnet dasturi yordamida transport agenti bilan bog‘lanishi va xatni yuborishini ko‘rsatish mumkin.



*Foydalanuvchi transport agenti yordamida faqat xatni yuborishi mumkin. Xatni qabul qilish uchun kirish agenti yordamida amalga oshiriladi.*

Pochta transport agentiga bog‘lanish uchun 25 portini oshkor ko‘rsatish orqali telnet dasturi qo‘llaniladi.

```
$ telnet localhost 25  
Trying 127.0.0.1...  
Connected to localhost.
```

```
Escape character is '^]'.
220 master.linux.zukko.uz ESMTP Sendmail
8.12.10/8.12.10; Sun, 28 Mar 2016 17:32:41
+0400
```

Pochtani yuborishda bu jo'natma aynan kimdan yuborilayotganligini albatta ko'rsatish kerak bo'ladi. Buning uchun SMTP protokolining mail from komandasi qo'llaniladi. Jimlik holati bo'yicha transport agenti yuboruvchining adresida yuboruvchi e-mail-idagi domenni albatta ko'rsatishni talab qiladi.

```
mail from: ilhom@localhost
250 2.1.0 ilhom@localhost... Sender ok
```

Agar yuboruvchining pochta xabarida hammasi joyida bo'lsa, transport agenti tasdiqlaydi.

Endi rcpt to komandasi yordamida qabul qilib oluvchining adresi ko'rsatiladi.

```
rcpt to: root
250 2.1.5 root... Recipient ok
```

Data komandasi yordamida xatning matnini kiritish boshlanadi. Oxirgi satrda xatning yakunlanganligini bildiruvchi yagona belgi «.» bo'lishi shart.

```
data
354 Enter mail, end with "." On a line by
itself
```

```
Bello root
This is a letter from ilhom
```

«.» belgisi qo'yilgandan keyin transport agenti xatni yetkazib berish uchun qabul qiladi, bu haqda mos xabarni beradi.

```
250 2.0.0 i2SDWfx4003350 Message accepted
for delivery
```

Aloqa vaqtini yakunlash uchun quit komandasidan foydalaniladi.

```
quit
221 2.0.0 master.linux.zukko.uz closing
connection
Connection closed by foreign host.
```

Keltirilgan misoldan ko‘rinib turibdiki, transport agentiga pochta yetkazish uchun ikkita parametrni ko‘rsatish kerak: jo‘natuvchining adresi va qabul qiluvchining adresi, hamda xatning tanasi.

Xatning tanasida ko‘rgan sarlavha, masalan, From:, To:, X-Mailer; va boshqalarni transport agenti xatning tanasi sifatida qaraydi va to‘g‘ridan-to‘g‘ri qayta ishlamaydi. Bunday sarlavhalarni tekshirish yoki o‘zgartirish uchun sendmail sozlanishi mumkin.

Transport agenti xatni yetkazib berishda uning sarlavhasiga xatning transport agenti orqali o‘tganligini tasdiqlovchi satrlarni qo‘shadi.

```
Received:      from      localhost.localdomain
(localhost.localdomain [127.0.0.1])
  by master.linux.zukko.uz (8.12.10/8.12.10)
with SMTP id 12SDWfx4003350
  for root: Sun, 28 Mar 2004 17:33:17 +0400
```

Xat olib o‘tiladigan har bir transport agenti xuddi shunga o‘xshash satrlarni qo‘shadi. Bu satrlar orqali xatni o‘tish marshrutini aniqlash mumkin.

Yuqorida server va klientning o‘zaro aloqasiga doir yetarli darajada oddiy misol keltirilgan. Bu yerda kanalning shifrlashi va foydalanuvchining avtorizatsiyasi hisobga olinmagan.



1. Elektron pochta tizimini tashkil etuvchi asosiy komponentlarni sanab bering.
2. Transport agenti qanday funksiyalarni bajaradi?
3. Foydalanuvchi o‘zining pochta qutisiga kirishga ruxsatni qanday olishi mumkin?

## 6.3-§. Sendmail

Sendmail – bu UNIX tizimlarida eng ko‘p tarqalgan transport agenti hisoblanadi. U deyarli barcha Linux distributivlariga o‘rnatilgan, jumladan, Slackware Linux-ga ham. Sendmail agenti Erik Ollman tomonidan 1983 yilda yozilgan va keyinchalik turli dasturchilar tomonidan qayta ishlanib to‘ldirilgan.

Sendmail-da pochta xabarlarini bilan ishlashda xatning mazmunini tekshirishdan tashqari barcha amallarni bajaradi. Lekin, u xatni tekshirish uchun uni boshqa dasturlarga berishi mumkin. Odatda bunday tekshirishlarga pochta xabarini viruslarga tekshirish kiradi.

Ushbu mavzuda turli xil vaziyatlarda ishlashi uchun Sendmail-ni qanday konfiguratsiyalash bo‘yicha to‘liq to‘xtalib o‘tamiz.

**Sendmailning konfiguratsiya fayllari.** Dasturning asosiy konfiguratsiya fayli -sendmail.cf. Odatda bu fayl /etc/mail katologida joylashgan bo‘ladi. Faylda sendmail ning asosiy parametrlar va ishlash xususiyatlari aniqlanadi, jumladan, qo‘shimcha konfiguratsiya faylining parametrlari ham aniqlanadi, bu fayl ham odatda /etc/mail katologida joylashgan bo‘ladi.

Sendmail.cf faylidan tashqari tizimda submit.cf fayli ham bo‘lishi mumkin, bu fayl pochta uzatish agentining konfiguratsiya fayli hisoblanadi.

Sendmail yuklanishi bilan parametrlar bilan bog‘liq konfiguratsiya fayli sifatida keltirilgan fayllardan qaysi biri qo‘llaniladi (ya’ni, dastur qanday rejimda ishlaydi):

- -As – konfiguratsiya fayli submit.cf-ni qo‘llagan holda pochta uzatish agenti sifatida yuklash.

- -Am - sendmail.cf (jimlik holati bo‘yicha parametr) faylini qo‘llagan holda asosiy pochta serveri sifatida yuklash.

Submit.cf fayli odatda tahrirlanmaydi.

Agar sendmail.cf faylining mazmunini o‘qisak, quyidagi satrlarga o‘xshash ma’lumotlarni ko‘rishimiz mumkin:

```
R$* <@ $=w > $* $: $1 < @ $2 . > $3
```

```
R$* < @ $=M > $* $: $1 < @ $2 . > $3
```

```
R$* < @ $={VirtHost} > $* $: $1 < @ $2.>$3
```

Fayl formati dastlab faylni sintaksis tahlilni osonlashtirish uchun yaratilgan va shuning uchun ham oddiy foydalanuvchilar uchun tushunarli bo‘lmaydi. Odatda administratorlar juda ham kam



holatlarda sendmail.cf faylini tahrirleydilar. Faylni yaratish uchun uni yaratish va o'zgartirishlar kiritishni osonlashtiradigan maxsus m4 preprotssessorining makroslari qo'llaniladi.

**m4 preprotssessori yordamida sendmail-ni konfiguratsiyalash.** Oldingi bo'limda aytilganidek, m4 preprotssessori administratorning sendmail.cf va submit.cf konfiguratsiya fayllarini yaratish va o'zgartirish jarayonlarini osonlashtiradi.

Bu fayllarni hosil qilish uchun m4-sendmail.cf preprotssessorining makroslaridan tashkil topgan qo'shimcha paket o'rnatilishi hamda m4 preprotssessorining o'rnatilganligi tekshirib ko'rilishi kerak.

Makroslar /usr/share/sendmail/cf<sup>9</sup> katologida joylashadi. Bu katologda barcha makroslar bo'yicha haqidagi README faylini ham ko'rishingiz mumkin. Makroslarning o'zi esa qismkatologlarda joylashgan bo'ladi.

cf katologida UNIX-ning har xil versiyalari uchun sendmail.cf faylini hosil qilishda qo'llaniladigan makroslar andozasi (andozalari) mavjud.

sendmail.cf faylini olish uchun qo'llaniladigan makroslar tavsiflanadigan faylni yaratish kerak bo'ladi. Keyin uni preprotssessordan o'tkazish orqali konfiguratsiya faylini olish mumkin.

**m4 my.mc > /etc/mail/sendmail.cf**

m4 preprotssessorida qat'iy sintaksisga amal qilinishi kerak. Ixtiyoriy ortiqcha probel, noto'g'ri qo'yilgan qavs xatolik kelib chiqishiga yoki faylning noto'g'ri hosil qilinishiga sabab bo'lishi mumkin.

Izohlarning boshlanishi sifatida dnl ko'rsatmasidan (instruksiya) foydalaniladi. Ushbu ko'rsatmadan keyin satr oxirigacha yozilgan barcha yozuvlar izoh hisoblanadi. Har satrning oxirida makrosning oxirida tasodifiy probelni inkor qilish uchun dnl ni yozish tavsiya etiladi.

Agar parametr yoki satrni ko'rsatish zarur bo'lsa, u holda ular qo'shtirnoq ichida yozilishi kerak. Ochiluvchi qo'shtirnoq "" (yo harfi joylashgan tugmadagi), yopiluvchi qo'shtirnoq esa to'g'ri apostrof belgisi "" ekanligiga alohida e'tibor berish kerak.

Shuningdek, preprotssessor tilidagi faylda divert ko'rsatmasidan ham foydalanish mumkin. Bu ko'rsatma yordamida birdaniga bir nechta satrlarda izoh yozish mumkin bo'ladi.

---

<sup>9</sup> Бошқа дистрибутивларда /usr/share/sendmail-cf катологида жойлашган бўлиши мумкин.

```

divert(-1)
#
# Copyright (c) 1998-2002 Sendmail, Inc.
# Copyright (c) 1983 Eric P. Allman.
# Copyright (c) 1988, 1993
# The Regents of the University of
California.
divert(0)

```

divert(-1) – makros buferidagi oldingi urinishlarda qolgan ma'lumotlarni tozalaydi. divert(0) – makros boshlanishini belgilash uchun qo'llaniladi.

Sendmail-cf paketi bilan o'rnatilgan makroslardan foydalanish uchun, ularni include direktivasi yordamida birlashtirish kerak bo'ladi.

```
include (` ../m4/cf. m4' )dnl
```

**VERSIONID makrosi.** VERSIONID makrosi hosil qilinadigan konfiguratsiya faylini identifikatsiya qilish uchun qo'llaniladi. Izoh belgisidan keyin ko'rsatilgan parametr sifatida ko'rsatiladiganlarning barchasi o'zgartirilmasdan chiquvchi faylga joylashtiriladi.

```
VERSIONID (`My super mail server' )dnl
```

**OSTYPE makrosi.** Turli UNIX-larda sendmailning qo'shimcha konfiguratsiya fayllarini joylashtirish o'rni va nomi qabul qilingan. Har bir faylni define direktivasi yordamida alohida aniqlash mumkin. Lekin, bu direktivalar har safar mc faylida qayta tavsiflanmaydi, OSTYPE makrosi qo'llaniladi.

```
OSTYPE (linux) dnl
```

Makrosda parametr sifatida /usr/shane/sendmail/cf/ostype katologida joylashgan fayl nomi kengaytmasiz ko'rsatiladi. Faylda jimlik holati bo'yicha UNIX ning har bir turi uchun parametrlar aniqlanadi.

linux.m4 faylining mazmuni:

```

VERSIONID(`$Id: linux.m4,v 8.13 2000/09/17
17:30:00 gshapiro Exp $')
define(`confEBINDER', `/usr/sbin')
ifdef(`PROCMail_MAILER_PATH', ,
define(`PROCMail_MAILER_PATH',
`/usr/bin/procmail'))
FEATURE(local_procmail)

```

**DOMAIN makrosi.** Agar bir xil parametrli bir nechta pochta serverlari mavjud bo'lsa, ushbu parametrlar tavsiflanadigan /nsr/share/sendmail/cf/domain katologida fayl yaratish mumkin.

Keyin DOMAIN makrosi yordamida ushbu faylga bog'lanish kerak.

**DOMAIN (my=domain) dn1**

Domain katologidagi generic.m4 fayliga misol:

```
VERSIONID(`$Id:      generic.m4,      v      8.15
1999/04/04 00:51:59 ca Exp $')
define(`confFORWARD_PATH',
`$z/.forward.$w+$h:$z/.forward+$h:$z/.forwa
rd.$w:$z/.forward') dn1
define(confMAX_HEADERS_LEN, `2768') dn1
FEATURE(`redirect') dn1
FEATURE(`use_cw_file') dn1
```

**MAILER makrosi.** Bu makros sendmail qanday qilib pochta yetkazib berishini aniqlash uchun qo'llaniladi.

Xuddi shunday bundan oldingi makrosga o'xshash mailer uchun maxsus katolog mavjud bo'lib, bu katologda sendmail foydalanadigan pochta yetkazib berish agenti tavsiflanadi.

Misol uchun, agar pochta lokal pochta qutisi va smtp protokli bo'yicha transport agentiga yetkazish kerak bo'lsa, bu imkoniyatlarni tavsiflash kerak bo'ladi:

**MAILER (local) dn1**

**MAILER (smtp) dn1**

Haqiqatan, mailer smtp birdaniga bir nechta transport agentlarida qo'shiladi: smtp, esmtp, dsmtpl, smtp8 va relay.

Agar pochta lokal pochta qutisiga yetkazishda filtrlash talab etilsa, procmail dasturini qo'llash tavsiya etiladi.

**MAILER (procmail) dn1**

**FEATURE makrosi.** Bu makros pochta serverining turli xil xususiyatlarini tavsiflash uchun qo'llaniladi. Aksariyat parametrlar aynan shu makros yordamida aniqlanadi. FEATURE makrosi haqida to'liq ma'lumotlarni sendmail konfiguratsiyasida qarab chiqamiz.

Ko'rib o'tilgan makroslardan tashqari boshqa turdagi makroslar ham mavjud.

**m4 preprotssessor faylida makroslarni tavsiflash tartibi.** Fayllarni tavsiflash tartibi muhim ahamiyatga ega. Asosiy e'tiborni

MAILER makrosiga qaratish kerak. Bu makros faylning oxirida, lekin turli xil lokal konfiguratsiyalar tavsifidan oldin joylashishi shart.

Quyida makroslarni tavsiflash tartibi keltirilgan:

- VERSIONID
- OSTYPE
- DOMAIN
- local macro definitions
- FEATURE
- MAILER
- LOCAL\_CONFIG
- LOCAL\_RULE\_\*
- LOCAL\_RULESETS

#### 6.4-§. Sendmail-ning ish rejimi

Sendmail bir nechta ish rejimlarini qoʻllaydi. Ish rejimlarini –b parametri yordamida yuklashda koʻrsatish mumkin.

Parametrlar	Qiymati
-bd	Dastur demon rejimida ishlaydi. Tekshirish uchun 25-port ochiladi.
-bD	Yuqoridagi parametrga oʻxshash, lekin dastur foreground rejimida ishlaydi. Otladka vaqtida qabul qilinadi.
-bh	hoststat ga oʻxshash. Statistik maʼlumotlarni tasvirlaydi.
-bN	Eski statistik maʼlumotlarni oʻchiradi. Purgestat dasturiga oʻxshash
-bi	Pseudonimlarni oʻrnatadi. Newaliases dasturiga oʻxshash
-bm	Pochtani oddiy yetkazish rejimi. Bu qiymat jimlik holati boʻyicha qabul qilinadi.
-bp	Joʻnatishda navbatning mazmunini koʻrsatadi. Mailq dasturiga oʻxshash.
-bs	Dastur maʼlumotlarni standart kirishdan oʻqiydi, 25-portdan emas.
-bt	Adreslarni tekshirish rejimi. Faqat otladka vaqtida qoʻllaniladi.

**Dasturni yuklash va boshqarish.** Konfiguratsiya fayli

sendmail.cf-ni o'zgartirgandan keyin barcha o'zgarishlar dastur o'qishi uchun kuchga kiradi. Turli xil sendmailni tavsiflovchi adabiyotlarda dasturga HUP signalini yuborish taklif etilgan bo'lsada, bu yetarli emas, yaxshi usullardan biri – bu dasturni qayta yuklash hisoblanadi.

Qayta yuklash “qo'lda” bajarilishi yoki yuklovchi skript yordamida amalga oshirilishi mumkin.

Slackware Linux-ning yuklovchi skriptida sendmail-ni yuklash uchun /etc/rc.d/rc.sendmail yuklovchi skripti qo'llaniladi. Bu skriptda dasturni yuklovchi o'zining sendmail\_start() funksiyasi mavjud.

```
/usr/sbin/sendmail -L sm-mta -bd -q25m
```

```
/usr/sbin/sendmail -L sm-msp-queue -Ac -q25m
```

-bd parametri dasturning bu nusxasi 25-portni tekshirish uchun ochishi va transport agenti funksiyasini bajarishini bildiradi. Ya'ni, u pochta serveri bo'ladi.

-As parametri dasturning bu nusxasi 587-portni tekshirish uchun ochishi va uzatish agenti funksiyasini bajarishini bildiradi.

-q parametri sendmail jo'natishdagi navbatni ko'rish davriyligini va biriktirilgan xatlarni jo'natishga harakat qilishini aniqlaydi. 25m – 25 minutga kechikishni anglatadi.

Dasturning to'xtashi TERM signalini yuborish orqali amalga oshiriladi:

```
killall sendmail
```

Madomiki TERM signali dastur ishini to'xtatar ekan, sendmail ning o'zi barcha jarayonlarni to'xtatadi va zarur ma'lumotlarni saqlaydi.

## **6.5-§. Foydalanuvchining pochta qutisi**

Jo'natmalarni pochta qutisida saqlashning bir nechta ko'rinishlari mavjud:

- file based tizimi, har bir foydalanuvchi uchun uning jo'natmalari joylashadigan alohida fayl hosil qilinadi.

- directory based tizimi, uy katologida foydalanuvchilar uchun har bir jo'natma alohida faylga joylashadigan alohida katolog hosil qilinadi.

Sendmail jo'natmalarni saqlashning birinchi variantidan foydalanadi. Foydalanuvchilarning barcha pochta qutilari

/var/spool/mail katologida joylashadi. Lekin unda ikkinchi variantni ham sozlash mumkin. Bunday vaziyatda procmail dasturining xizmatlaridan foydalanishga to'g'ri keladi.

Pochta qutisini yaratish uchun standart dasturlar yordamida tizimga foydalanuvchi qo'shish kerak. Foydalanuvchilarda o'zining pochta qutisiga kirish uchun u o'zini tizimda identifikatsiya qilishi va parol o'rnatilgan bo'lishi kerak.

Odatda serverda pochta qutisi bo'lgan foydalanuvchilarga tizimga ssh orqali (login) kirishga ruxsat berish zarurati yo'q. Shuning uchun ham ularga shell-ning o'rniga (/etc/passwd faylining yettinchi maydoni) /bin/false dasturi yoki agar sabablarni tushuntirish zarur bo'lsa, oddiy skript yoziladi. Masalan, quyidagicha:

```
#!/bin/bash
echo; echo; echo
echo "*****"
echo "Hurmatli $LOGIN. Sizga tizimga kirish
taqiqlangan"
echo "Administrator"
echo "*****"
sleep 15
```

Foydalanuvchi tizimga kirishga urinib ko'rgan vaqtida uning ekraniga mos xabar chiqariladi va u 15 sekund davomida o'qish imkoniyatini beradi. Keyin dastur o'z ishini yakunlaydi va foydalanuvchi tizimdan chiqib ketadi.

Foydalanuvchi uchun uy katologi hosil qilish kerakmi? Bu savolga javob pochta saqlash uchun qaysi tizimdan foydalanishingiz va foydalanuvchining pochta qutisiga kirish uchun qo'llaniladigan protokollarga bog'liq.

POP3 protokolidan foydalanilganda uy katologini yaratish shart emas. Agar pochta qutisiga kirish uchun IMAP protokolidan foydalanilsa, u holda bunday katolog bo'lishi shart. IMAP protokolidan foydalanilganda foydalanuvchining kiruvchi pochta larini uning uy katologida saqlash tavsiya etiladi.

## 6.6-§. Yuborishda navbat bilan ishlash

Sendmail-da pochta yetkazishning to'rtta rejimi mavjud:

- Fon rejimi. Maktublar darhol yetkazib beriladi. Yetkazish

uchun sendmail yangi alohida jarayon hosil qiladi.

- Interaktiv rejim. Maktub darhol yetkaziladi, lekin uni yetkazish uchun bitta va aynan shu jarayon shug'ullanadi.

- Navbatga qo'yish rejimi. Maktublar jo'natish uchun navbatga qo'yiladi, keyin qaerga yuborishni navbatni qayta ishlovchilar har qiladi.

- Kechiktirish rejimi. Bu rejim ham navbat rejimiga o'xshash, lekin navbatdagi barcha qidiruv amallari jadvalga tushiriladi, DNS-da va h.k.

Jimlik holati bo'yicha sendmail fon rejimidan foydalanadi. Ya'ni u maktublarni darhol yetkazishga harakat qiladi. Lekin, agar darhol yetkazish imkoni bo'lmasa yoki tizim juda katta yuklanishda bo'lsa, maktublar jo'natish uchun navbatga tushadi.

Uzatishda navbatga tushgan maktub /var/spool/mqueue katologida joylashadi. Pochtani uzatish agentining maktublarni uzatish uchun o'z navbati bo'ladi - /var/spool/clientmqueue.

Uzatish navbatining mazmunini ko'rish uchun mailq komandasidan foydalanish mumkin.

Uzatishdagi navbatni darhol ko'rib chiqish (tahlil qilish) uchun – q parametri bilan sendmail-dan foydalaniladi. Bunday vaziyatda dastur navbatda turgan maktublarning barchasini jo'natishga harakat qiladi. Agar maktub jo'natilmay qolsa, uzatish navbatida qoladi. Barcha navbatni ko'rib chiqqandan keyin sendmail ning bu nusxasi o'z ishini yakunlaydi.

Ba'zan, shunday vaziyat bo'ladiki, navbatdagi maktublarni uzatish boshlanganda sendmail ishi sekinlashadi. Bunday holatda boshqacha yo'l tutish mumkin:

- Navbatni boshqa katologga ko'chirish. Sendmail-ning asosiy nusxasi me'yordagi tezlikda o'z ishini davom ettiradi. Yangi katologdagi maktublarni yuborish uchun sendmail-ning alohida nusxasini quyidagi parametr bilan ishga tushiriladi:

-q -O QueueDirectory=new\_queue\_dir.

- Agar yuborishdagi navbatning to'lishi holati tez-tez yuz bersa, u holda uni bir nechta guruhga ajratish va QUERY\_GROUP makrosidan foydalanish kerak.

- Agar jo'natma birinchi urinishda amalga oshmasa, jo'natmani boshqa kompyuterga o'tkazuvchi FALLBACK\_MX parametrini aniqlash.

- jo'natma yuboriladigan har bir kompyuter haqidagi

ma'lumotlar saqlanadigan katologni ko'rsatuvchi HOST\_STATUS\_DIRECTORY parametrini aniqlash. Natijada uzatish navbatidagi ko'p sonli maktublar joylashib qolgan serverning unumdorligi sezilarli darajada oshadi.

### 6.7-§. Log tizimi

Sendmail xatoliklar va o'zining holati haqidagi barcha xabarlarni mail vositasi yordamida Syslog tizimiga jo'natadi.

Barcha xabarlar sendmail satri bilan boshlanadi. Agar sendmailning bir nechta jarayonlari, masalan, asosiy demon jarayoni va pochmani uzatish agenti yuklangan bo'lsa, -L parametri yordamida dastur xabarlari qanday so'z bilan boshlanishini aniqlash mumkin.

Slackware Linuxning yuklash skripti sendmailning yuklanishida unga -L sm-mta yoki -L sm-smp-queue parametrini beradi. Bundan kelib chiqadiki, transport agentidan uzatilgan barcha xabarlar sm-mta so'zi bilan boshlanadi, pochmani uzatish agenti tomonidan Syslog tizimiga yuborilgan barcha xabarlar esa sm-smp-queue so'zi bilan boshlanadi.

Jimlik holati bo'yicha Syslog tizimi barcha xabarlarini /var/log/maillog fayliga tushadigan qilib sozlanadi. Ushbu faylga uzatilgan va qabul qilingan maktublar, uzatish vaqtida paydo bo'lgan xatoliklar, dasturni yuklanishi haqidagi barcha ma'lumotlar kelib tushadi.

Masalan, yuklanishda faylga quyidagi satrlardan iborat xabar kelib tushadi:

```
Mar 31 9:14:22 home sm-mta(1090): starting
daemon (8.12.16): SMTP+queueing@01:00:00
```

```
Mar 31 9:14:22 home sm-smp-queue(1099):
starting daemon (8.12.16): queueing@01:00:00
```

Maktubni log fayliga yuborganda bir nechta satrlar kelib tushadi:

```
Mar 30 10:57:31 home sm-mta(2016):
i2U6vVW6002005: from=root, size=4707, class=0,
nrcpts=1
msgid=<201603300657.i2U6vVW6002016@master.linux
.zukko.uz >,
relay=root@localhost
Mar 30 10:57:31 home sm-mta(2016):
i2U6vVq4002016:
```



```
from=<root@master.linux.zukko.uz >, size=4991,  
class=0, nrcpts=1  
msgid=<201603300657.i2U6vVW6002016@master.linux  
.zukko.uz >,  
proto=ESMTP, daemon=MTA,  
relay=localhost.localdomain [127.0.0.1]  
Mar 30 10:57:31 home sm-mta(2016):  
i2U6vVq6002016: to=root, ctladdr=root (0/0),  
delay=00:00:00, xdelay=00:00:00, mailer=relay,  
pxi=34707, relay=[127.0.0.1], dsn=2.0.0,  
stat=Sent (i2U6vVq4002016 Message accepted for  
delivery)  
Mar 30 10:57:31 home sm-mta(2017):  
i2U6vVq4002016: to=<root@master.linux.zukko.uz  
>, ctladdr=<root@master.linux.zukko.uz > (0/0),  
delay=00:00:00, xdelay=00:00:00, mailer=local,  
pri=35212, dsn=2.0.0, stat=Sent
```

Ushbu satrlardan maktubni yuborish holati va server ish jarayonida qanday xatoliklar yuz berganligini aniqlash mumkin.



1. Sendmail dasturining konfiguratsiya fayli qanday nomlanadi va u qaerda joylashgan bo‘ladi?
2. Sendmail dasturida qanday makroslar yordamida pochtani uzatish usullarini aniqlash mumkin?
3. Sendmail dasturini qaysi parametr 25-portni oshkor tekshirish bilan demon rejimida ishlashiga majbur qiladi?
4. Foydalanuvchining pochta qutisi qaerda joylashadi?
5. Yuborish navbatini qaysi komanda yordamida ko‘rish mumkin?
6. Yuborish navbatida turgan maktublarni darhol uzatish uchun sendmailni qanday parametr bilan yuklash kerak?

## 6.8-§. Oddiy pochta serverini sozlash

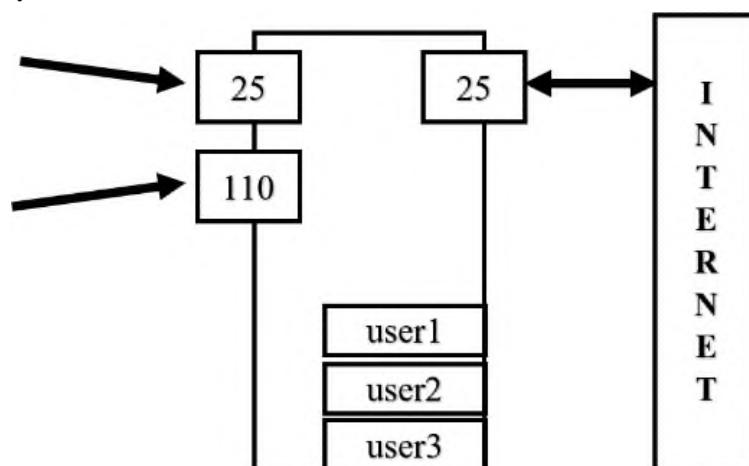
Birinchi misol sifatida bir nechta pochta qutilariga xizmat ko'rsatadigan oddiy pochta serverini sozlashni ko'rib chiqamiz.

Paketdan sendmail-ni o'rnatgandan keyin, u ishlashi uchun bir qancha cheklovlarga ega bo'ladi:

- Pochtalarni faqat u o'rnatilgan adreslangan kompyuterlardan qabul qiladi.

- Faqat bevosita serverda ishlaydigan foydalanuvchilarga maktub jo'natishga xizmat qiladi.

- 25-portdagi so'rovlarni tinglaydi (tekshiradi), lekin faqat lo interfeysida<sup>10</sup>.



Faraz qilaylik, sendmail o'rnatilgan kompyuterning nomi smtp.samtuit.uz bo'lsin. Ushbu kompyuterda user1 foydalanuvchisi ro'yxatdan o'tgan. samtuit.uz domeniga mas'ul bo'lgan DNS serverda MX yozuvi mavjud:

```
samtuit.uz. IN MX 5 smtp.samtuit.uz .
```

Bu yozuv barcha pochta serverlariga pochta samtuit.uz domeniga mo'ljallanganligini (masalan, user1@samtuit.uz), va pochtalarni smtp.samtuit.uz joylashgan kompyuterdagi pochta serveriga yuborish zarurligini ko'rsatadi.

Jimlik holati bo'yicha sendmail u o'rnatilgan kompyuterga mo'ljallangan jo'natmalarni qabul qiladi, ya'ni e-mail pochta qutisi user1@smtp.samtuit.uz ko'rinishida berilgan jo'natmalarni oladi. Pochta qutisi nomi user1@class.unix kabi ko'rsatilgan jo'natmalarni serverga qabul qilmaydi.

<sup>10</sup> Slackware Linux-дан фаркли дистрибутивларда

Ikkinchi cheklov: jimlik holati bo'yicha server orqali pochta jo'natmalarini (RELAY) qabul qilish taqiqlanadi. Bu cheklov serverni himoyalash uchun mo'ljallangan. Agar sendmail ruxsat berilgan jo'natmalar (dasturning eski versiyalaridagi kabi) o'rnatilgan bo'lsa, u holda Internetning ixtiyoriy foydalanuvchisi pochta jo'natmalarini yuborish uchun undan foydalanishi mumkin bo'ladi. Spam tayyorlovchilar tarmoqda ochiq RELAY bilan serverni qidiradi va ulardan spamlarni yuborishda foydalanadilar. Agar pochta serveri log fayllarini ko'rsak, ushbu server orqali maktublarni yuborishga urinishlar va mos xatoliklar haqidagi xabarlarni o'qishimiz mumkin.

Lekin muammo spam tayyorlovchilarida ham emas, spam tayyorlovchilar haqida ma'lumotlarni yig'uvchi DNSBL serverlarida. Agar Sizing serveringiz ushbu serverlar qatoriga tushib qolsa (agar Siz Internet dan jo'natmalarni yuborishga taqiq qo'ymagan bo'lsangiz u albatta tushadi), u holda xuddi shunday xizmatlardan foydalanuvchi boshqa pochta serveri, Sizing serveringizga yuborilgan maktublarni qabul qilishga o'tadi.

## 6.9-§. Konfiguratsiya fayllarini hosil qilish

Konfiguratsiya faylini hosil qilish uchun standart makroslar VERSIONID, OSTYPE, FEATURE va MAILER-lardan foydalaniladi.

Sendmail dasturining barcha asosiy xususiyatlari FEATURE makrosi yordamida tavsiflanadi. Biz sendmail-ni linux.zukko.uz domeni uchun jo'natmalarni qabul qilishga majburlashimiz va bizning tarmoqning ichki klientlariga jo'natmalarni yuborishga ruxsat berishimiz kerak.

**use\_cw\_file vositasi.** Sendmail jo'natmalarni qabul qiladigan domen nomini /etc/mail/local-host-names faylidan izlaydi.

**FEATURE (use\_cw\_file)dnl**

/etc/mail/local-host-names faylida bitta satrga alohida domen nomini yozishimiz kerak. Masalan:

**linux.zukko.uz**



*Oxirgi satr albatta bo'sh qoldirilishi kerak! Agar buni bajarmasak, u o'qilmay qoladi va sendmail oxirgi satrda tavsiflangan domen uchun jo'natmalarni qabul qilmaydi.*

**Retranslyatsiya** (qayta yuborish). Ichki tarmoqda joylashgan

klientlar bizning server orqali jo‘natmalarni uzatishiga ruxsat berishimiz kerak.

Jo‘natmani retranslyatsiya qilishga ruxsat berishning ko‘plab usullari mavjud. Eng ommalashganlaridan biri access\_db vositasini qo‘llashdir. Bu vosita sendmail maxsus kirish ma’lumotlar bazasidan foydalanishga imkon beradi, bu baza xesh (hash) jadvali sifatida ishlaydi.

```
FEATURE ( `access_db` ,          `hash          -T<TMPF>  
/etc/mail/access.db' ) dn1
```

access\_db vositasini aniqlash uchun kirish bazasi saqlanayotgan access.db fayli ko‘rsatiladi.

access.db – o‘zida hash jadvalini saqlab turuvchi binar fayl hisoblanadi. Ushbu faylni hosil qilish yoki unga o‘zgartirishlar kiritish uchun, dastlab makemap dasturi yordamida binar faylga aylantiriladigan matnli fayl hosil qilinadi.

Odatda access matnli faylining manbasi access.db joylashgan katologda joylashgan bo‘ladi.

Xesh jadval unikal qidiruv kalitlari va qiymatlaridan tashkil topgan bo‘ladi, shuning uchun ham access faylida birinchi kalit probel yoki tabulyatsiyadan keyin kalitga mos keladigan qiymat yoziladi.

Kalit sifatida quyidagilardan foydalanish mumkin:

- IP adreslar, jumladan to‘liq bo‘lmagan IP adreslar. Masalan, 172.20.21.1 yoki 172.20.21.
- domen nomlari. Masalan, samtuit.uz.
- E-mail. Masalan: ilhom@samuit.uz.
- e-mail ning to‘liq bo‘lmagan qismlari. Masalan, ilhom@
- Teglar: Connect, From, To, Spam. Masalan, To: samtuit@

Qiymatlar maydonida turli kalit so‘zlaridan foydalanish mumkin, lekin hozirgi vaziyatda biz jo‘natmani yuborishga ruxsat berishga yordam beradigan RELAY kalit so‘zi bilan cheklanamiz.

Jo‘natmani yuborishga ruxsat berish uchun access faylida quyidagi satrlarni joylashtirishimiz kerak bo‘ladi:

```
Connect:127.0.0.1 RELAY  
Connect:172.20.21 RELAY
```

Oxirgi satr jo‘natmani 172.20.20/24 tarmog‘ida joylashgan barcha kompyuterlarga yuborishga ruxsat beradi.

Access faylini access.db fayliga o‘zgartirish uchun quyidagi dasturni ishga tushirish kerak:

## **makemap hash access < access**

Makemap dasturi hash jadvalini saqlovchi faylni hosil qiladi. Bu fayl access.db deb ataladi (fayl kengaytmasini oshkor ravishda ko'rsatmaslik ham mumkin). Ma'lumotlar access faylidan olinadi. Keltirilgan misolda faylga yo'l to'liq ko'rsatilmaganligiga e'tibor bering, bunda makemap dasturini /etc/mail katologida ishga tushirish kerak yoki access.db va access fayllarga yo'lni to'liq ko'rsatish lozim bo'ladi.



1. Qaysi makros yordamida local-host-names faylini qo'llashni qo'shish mumkin?
2. Qanday dastur yordamida kirish bazasi hosil qilinadi? Bajarilishi kerak bo'lgan komandalar satrini yozing.
3. Kirish bazasining qaysi satriga IP 172.20.21.1 kompyuteriga uzatish ruxsati qo'shiladi?



- my.mc faylida access\_db vositasini tavsiflash kerak.
- /etc/mail/access fayliga pochta serveri orqali klient kompyuterlaridan jo‘natmalarni uzatishga ruxsat beruvchi satrni qo‘shish kerak.
- access faylidan access.db faliga hash jadvallarini hosil qilish.
- Pochta serveriga foydalanuvchilarning qayd yozuvchilari (учетные записи)ni qo‘shish.

Ko‘rib turganimizdek, yuqorida sanab o‘tilgan amallar oddiy pochta serverini sozlash amallaridan hech qanday farq qilmaydi. Katta istisno bilan local-host-names fayliga a.uz domeni qo‘shildi. Bu pochta serveri nafaqat samtuit.uz domenidan balki, a.uz domeni uchun ham jo‘natmalarni qabul qiluvchi serverga aylanganligini anglatadi.

Faraz qilamiz, klient uning kompaniyasi direktorining e-mail adresi boss@a.uz bo‘lishini so‘radi. Bizda esa samtuit.uz kompaniyasining boss foydalanuvchisi boss@samtuit.uz nomli e-mail bilan mavjud. local-host-names fayliga a.uz yozuvchi yozilgandan keyin boss@a.uz va boss@class.unix adreslaridan yuborilgan barcha maktublar bitta va aynan boss foydalanuvchisining pochta qutisiga kelib tushadi. @ belgisidan keyin yoziladigan e-mailning qismi maktubni qaysi kompyuterga yetkazish zarurligini ko‘rsatadi. boss@a.uz e-mail-idan yuborilgan maktub boss foydalanuvchisining pochta qutisiga emas, balki boss\_a pochta qutisiga tushishi uchun nima qilish kerak?

Bu muammoni yechishning eng yaxshi usuli pochta serverining virtual xositingi yordamida osongina tashkil etiladigan virtual foydalanuvchilar jadvalini qo‘llaydigan virtusertable vositasidan foydalanish hisoblanadi.

- my.mc fayliga virtusertable vositasini qo‘shamiz.

```
FEATURE(`virtusertable', `hash -o  
/etc/mail/virtusertable.db') dn1
```

Bu yozuv bitta satrda yozilishi shart.

- boss\_a foydalanuvchilarining lokal pochta qutilari va boss@a.uz e-mail-dan yuboriladigan barcha jo‘natmalarni uzatishga sendmailni majburlovchi satrni qo‘shish uchun /etc/mail/virtusertable faylini hosil qilamiz.

```
boss@a.uz      boss_a
```

- virtusertable faylidan makemap dasturi yordamida

virtusertable.db hash jadvalini hosil qilamiz.

- my.mc faylidan sendmail.cf faylini hosil qilamiz.
- Pochta serverini qayta yuklaymiz.

Klient kompyuterida u boss\_a jo‘natmalarini qaerdan olishini aniqlash uchun pochta qutisi nomini ko‘rsatish kerak. Pochta adresining Mail From maydoniga boss@a.uz adresining a.uz domenini ko‘rsatish kerak bo‘ladi.

### **6.11-§. Doimiy Internet-ga ulangan tashkilot uchun pochta serverining virtual xostingini tashkil etish**

Faraz qilaylik, bizga pochta serveri doimiy Internet-ga ulangan, lekin ishlamayotgan klient (masalan, b.uz) murojaat qildi. Masala shundan iboratki, b.uz domeni uchun mo‘ljallangan barcha jo‘natmalarni uning pochta serveri smtp.b.uz ga qabul qilish va undan uzatishni sozlab berishimiz kerak. Klientning pochta serverini unga kiruvchi barcha jo‘natmalar bizning server orqali o‘tishiga mos sozlab berishimiz kerak.

Ushbu masala yechish uchun smtp.class.unix serverida bajariladigan amallar:

- b.uz domenining hududni tavsiflash faylida MX yozuvini smtp.samtuit.uz pochta serveriga havola qilingan bo‘lishi.

- Serveringizning sendmail.cf fayli tomonidan hosil qilingan my.mc fayliga use\_cw\_file vositani qo‘shish.

- Serveringiz b.uz domeniga yuborilgan jo‘natmalarni qabul qilishi uchun /etc/mail/local-host-names fayliga ushbu domenning nomini qo‘shish.

- my.mc faylida access\_db vositasini tavsiflash.

- /etc/mail/access fayliga smtp.b.uz pochta serveriga yuborishga ruxsat satrini qo‘shish.

- access faylidan access.db fayliga hash jadvalni hosil qilish.

- my.mc fayliga virtusertable vositasini qo‘shish.

- /etc/mail/virtusertable fayliga smtp.b.uz kompyuteridan b.uz domenga keluvchi barcha jo‘natmalarni sendmail qayta yuborishini ta’minlovchi satrni qo‘shish. %1 belgisi, jo‘natma aynan shu foydalanuvchiga ekanligini bildiradi.

**@b.uz %1@smtp.b.uz**

- virtusertable faylidan virtusertable.db fayliga hash jadvalni hosil qilish.



- Pochta serverini qayta yuklash.

Bizning server b.uz domeniga mo'ljallangan barcha jo'natmalarni smtp.b.uz serveriga yuborishi uchun virtusertable dan foydalanish kerak.

Klientning pochta serverini oldingi laboratoriya ishida bajarilgandek sozlaymiz. Ya'ni, foydalanuvchilarning barcha pochta qutilari ushbu serverda joylashadi, foydalanuvchilarning so'rovlarini mustaqil qayta ishlaydi.

Sendmail jo'natmalarni uzatishda jo'natmalarni mustaqil ravishda belgilangan serverga yetkazishga harakat qiladi. Masala sharti bo'yicha u barcha kiruvchi jo'natmalarni bizning server orqali uzatilishini ta'minlashimiz kerak. Buning uchun smtp.b.uz serverining mc faylida SMART\_HOST deb ataluvchi vositani aniqlashimiz kerak bo'ladi.



1. Virtusertable vositasi nima uchun mo'ljallangan?
2. Virtual foydalanuvchi jadvaliga b.uz domeniga mo'ljallangan barcha jo'natmalar b\_user foydalanuvchisining lokal pochta qutisiga kelib tushishi uchun qanday satrni qo'shish kerak bo'ladi?

## 6.12-§. Sendmail-ning qo‘shimcha konfiguratsiyasi

Bu bo‘limda sendmail-ni sozlashning quyidagi shakllari qarab chiqiladi:

- Konfiguratsiya parametrlari.
- Xavfsizlik parametrlari.
- Kirish bazasi.
- Qora ro‘yxat.

**Konfiguratsiya parametrlari.** Sendmail-ning konfiguratsiya parametrlarini mc faylida define direktivasi yordamida berish mumkin.

Masalan,

```
Define( `confPRIVATE_FLAGS' ,  
`novrfy.noexpn' ) dn1
```

Sendmailda 175 ta konfiguratsiya parametrlari mavjud. Biz faqat eng ko‘p qo‘llaniladigan parametrlarini qarab chiqamiz.

Quyidagi jadvalda ko‘rsatilgan parametrlarda uni aniqlashda nomiga albatta qo‘shiladigan conf so‘zi mavjud emas

Parametrlar	Tavsifi
MIN_FREE_BLOCKS	Bu parametr bloklarda o‘lchanadigan disk maydonining bo‘sh bloklari sonini aniqlaydi. Agar diskda joy ko‘rsatilgan sondan kichik bo‘lsa, u holda sendmail jo‘natmalarni qabul qilishni to‘xtatadi. Uning qiymati jimlik holati bo‘yicha 100 ga teng.
MAX_MESSAGE_SIZE	Sendmail yuborilgan jo‘natmani qabul qilish uchun xatning baytlardagi maksimal o‘lchami Jimlik holati bo‘yicha qiymat cheklanmagan.
DELIVERY_MODE	Sendmailning 4 ta ish rejimidan bittasini tanlash parametri. Jimlik holati bo‘yicha qiymati background ga teng.
MAX_NOR	Xat o‘tishi mumkin bo‘lgan serverlarning maksimal sonini

	aniqlaydi. Jimlik holati bo'yicha qiymati 25 ga teng
LOG_LEVEL	Log fayliga qancha to'liq xabar yuborilishi kerakligini aniqlaydi. Jimlik holati bo'yicha qiymati 9 ga teng.
PRIVACY_FLAGS	SMTP protokoli metodlarini qo'llashdagi cheklovlarni aniqlaydi. Jimlik holati bo'yicha qiymati authwarningsga teng.
SORY_ERRORS_TO	Xatoliklar haqidagi xabarlar uzatiladigan e-mail ni aniqlaydi. Jimlik holati bo'yicha qiymati aniqlanmagan.
TO_nom	Bu parametr ko'rsatilgan vaziyat uchun timeout ning turli qiymatlarini beradi.
TO_IDENT	Yuboruvchining autentifikatsiyasi haqidagi so'roviga javob berishni kutish vaqti. Agar tarmoqda faqat Windows klientlar ishlatilayotgan bo'lsa, ushbu parametrga 0 qiymatni o'rnatish tavsiya etiladi. Jimlik holati bo'yicha qiymati 5s ga teng.
TO_QUEUEWARN	Oldin foydalanuvchiga ogohlantirish yuboradi va xat qancha vaqt navbatda turishini aniqlaydi. Jimlik holati bo'yicha qiymati 4h ga teng.
TO_QUEUERETURN	Xat qancha vaqt navbatda turishi mumkinligini aniqlaydi. Jimlik holati bo'yicha qiymati 5d ga teng.
QUEUE_LA	Navbatda jo'natish uchun o'rin almashadigan, darhol

	<p>yuborilmaydigan barcha xabarlarning o'rtacha yuklanish ko'rsatkichi.</p> <p>Jimlik holati bo'yicha qiymati 8*(protessorlar soni) ga teng.</p>
MAX_ALIAS_RECURSION	<p>Sendmail ruxsat berishi mumkin bo'lgan biriktirilgan psevdonimlarning maksimal sonini aniqlaydi. Jimlik holati bo'yicha qiymati 10 ga teng.</p>
MAX_DAEMON_CNILDREN	<p>Paydo bo'ladigan jarayonlarning maksimal soni. Sendmail jo'natmani yetkazib berish yoki qabul qilishda har bir bog'lanish uchun yangi nusxasini hosil qiladi. Jimlik holati bo'yicha qiymati cheklanmagan.</p>
MAX_NEADERS_LENGTH	<p>Xat sarlavhasining maksimal miqdori.</p> <p>MAX_MESSAGE_SIZE faqat xatning o'lchamini cheklaydi, uning sarlavhasi o'lchamini cheklamaydi. Jimlik holati bo'yicha qiymati 32768 ga teng.</p>
SMTP_LOGIN_MSG	<p>Klientning pochta serveriga bog'lanishida kiritishi mumkin bo'lgan satrlarni aniqlaydi. Jimlik holati bo'yicha qiymati \$j Sendmail \$v/\$Z;\$b ga teng.</p>
MAX_RCPTS_PER_MESSAGE	<p>Bitta xatni oluvchilar sonini cheklaydi. Jimlik holati bo'yicha qiymati cheklanmagan.</p>
DONT_PROBE_INTERFACES	<p>Ushbu parametr o'rnatilgan bo'lsa, sendmail ishga tushirilganda tarmoq interfeyslari nomini aniqlashga urinmaydi va ularni w sinfidagi konfiguratsiya fayllariga qo'yadi. Odatda bunday usul bilan sendmail xat</p>

	qabul qiladigan kompyuter nomlari aniqlaydi. Jimlik holati bo'yicha qiymati false-ga teng.
REJECT_MSG	Jo'natma rad qilinganda beriladigan xabarlarni aniqlaydi. Jimlik holati bo'yicha qiymati 550 Access denied-ga teng.
RELAY_MSG	Xatni yuborish taqiqlanganda beriladigan xabarlarni aniqlaydi. Jimlik holati bo'yicha qiymati 550 Relaying denied-ga teng

**Xavfsizlik parametrlari.** PRIVACY\_FLAGS parametri yordamida xavfsizlikning bir qancha parametrlarini aniqlash mumkin:

- Tashqi dunyoda tizim haqida qanday ma'lumotlar borligini SMTP protokoli bo'yicha bilish mumkin.

- Foydalanuvchilar jo'natma xabarlari navbatini ko'rishi yoki qayta ishlashi mumkinmi yoki yo'qligini.

- SMTP bog'lanishning oxiridagi qarama-qarshi tugunidan nima talab qilinishi.

Jadvalda ba'zi xavfsizlik parametrlari keltirilgan

<b>Qiymatlar</b>	<b>Tavsifi</b>
public	Xavfsizlik tekshiruvini bekor qilish. Yaxshi variant emas.
needmailhelo	Serverga bog'lanishda HELO komandasi yordamida bog'lanuvchini tanishtirish kerak.
noexpn	EXPN komandasini taqiqlash
novrfy	VRFY komandasini taqiqlash
needexpnhelo	HELO komandasini qo'llamasdan oldin EXPN komandasidan foydalanish taqiqlanadi
needvrfyhelo	HELO komandasini qo'llamasdan oldin VRFY komandasidan foydalanish taqiqlanadi
noverb	EXPN komandasining "batafsil" rejimini taqiqlaydi, bunda forward fayllaridan va psedonimlar bazasidan uzatish to'g'risida qo'shimcha ma'lumotlar beriladi.
restrictmailq	Faqat guruhdagi va /var/spool/mailq katologiga tegishli foydalanuvchi xabarlar navbatini ko'rishi mumkin.

restnetqrun	Faqat mqueue katologi egasi xabarlar navbatini qayta ishlashi mumkin.
noetrn	ETRN komandasi yordamida tugunlar jo‘natmalarni uzatishni boshlashini taqiqlaydi. Odatda, bu komanda dial-up tarmog‘i orqali ulangan tugunlar tomonidan ishlatiladi.
authwarnings	Xabarlarga “Authentication Warning” sarlavhasini qo‘shadi (jimlik holati bo‘yicha o‘rnatiladi)
noreceipts	Yetkazib berish holati kodlarini berish taqiqlanadi.
nobodyreturn	Yetkazib berish haqidagi holat kodini berishda xabarning tanasini qaytarmaydi
goaway	SMTP so‘rovlarining barcha statuvslarini inkor qilish: ETRN, VRFY va boshqalar

Xavfsizlik parametrlarini aniqlash vergul orqali bajariladi. Masalan,

```
define ( `confPrIVACY_FLAGS` ,
`authwarnings , novrfy , noexpn , restictqrun` ) dn1
```

**Kirish bazasi.** Kirish bazasini qo‘llashni ishga tushirish uchun access.db-dan foydalanish kerak. Biz buni aniqlaganmiz, lekin bazadan faqat jo‘natmalarni uzatishga ruxsat olish uchun foydalanganmiz.

Access\_db vositasini aniqlagandan keyin kirish bazasining qo‘shimcha imkoniyatlaridan foydalanishda qo‘l keladigan qo‘shimcha blacklist\_recipients vositasini aniqlash mumkin.

```
FEATURE (blacklist_recipients)dn1
```

Kirish bazasi, ikkita qism: qidiruvning unikal kaliti va qiymatidan iborat hash jadvallari ko‘rinishida bajariladi.

Qiymatlar maydoni quyidagi kalit so‘zlaridan foydalanish mumkin:

- RELAY – uzatishdagi jo‘natmani qabul qilish.
- REJECT – xatolik haqidagi namunaviy xabar bilan jo‘natmani inkor qilish.
- DISCARD – sabablarini tushuntirmasdan jo‘natmani o‘chirish.
- OK – jo‘natmani oddiy rejimda qabul qilish, hattoki, agar e-mail kiritilgan REJECT va DISCARD qoidalarining cheklovlari ostida

tushsa ham.

- FRIEND – spamlar bilan kurashish uchun xizmat qiladi. Uning ishlashi uchun quyidagicha oshkor ravishda tavsiflash kerak

**FEATURE(`delay\_checks', `friend').**

- HEATER – spamlar bilan kurashish uchun xizmat qiladi. Uning ishlashi uchun quyidagicha oshkor ravishda tavsiflash kerak:

**FEATURED(`delay\_checks', `heater").**

- xxx xabar – xat qabul qilinmaydi. Xatolik kodi va xabarini orqaga qaytaradi. xxx o‘rniga RFC821da aniqlangan xatolik kodi turishi shart.

- va boshqalar.

Kirish bazasi yordamida kiruvchi jo‘natmalarni cheklashga misollar.

@mail.ru 550 biz oshkor serverlar bilan jo‘natmalarni qabul qilmaymiz

**user@mail.ru OK**

**@spamer.com DISCARD**

**sex@ REJECT**

Birinchi satrda mail.ru-dan yuborilgan “550 biz oshkor serverlar bilan jo‘natmalarni qabul qilmaymiz” xabarining qaytarilishi bilan ixtiyoriy jo‘natmani qabul qilish taqiqlangan.

Ikkinchi satrda oldingi qoida – kelishuvni istisno qiladi. user@mail.ru foydalanuvchisidan u mail.ru domenida joylashganiga qaramasdan jo‘natma qabul qilinadi.

Uchinchi satrda spamer.com domenidan ixtiyoriy jo‘natmani qabul qilish taqiqlangan. Javobda xatolik xabari yuborilmaydi.

Oxirgi satrda ixtiyoriy domendagi sex foydalanuvchilaridan jo‘natmalar qabul qilinishi taqiqlangan. Javobda xatolik haqidagi standart xabar qaytriladi.

Kalit maydonida To:,FROM: va boshqa kalit so‘zlari bilan e-mailni ko‘rsatish mumkin.

**Qora ro‘yxat.** Sendmail-da mail-abuse.org saytida chop etiladigan MAPS (Mail Abuse Prevention System) deb ataluvchi qora ro‘yxatdan foydalanish mumkin. Shuningdek, ixtiyoriy ro‘yxat.

Ro‘yxatga bog‘lanish uchun quyidagi vositalardan foydalanish mumkin:

**FEATURE(dnsbl)**

**FEATURE(enhdnsbl)**

Bu vositalar sendmail-ni ro'yxatga kiritilgan domen yoki foydalanuvchilardan jo'natmalarni qabul qilmaslikka majburlaydi.

Ro'yxatning bir nechta ko'rinishdari mavjud:

- ma'lum spamerlarning IP adreslaridan iborat.
- ochiq retranslyatsiyani qo'llovchi tugunlar IP adreslaridan iborat.

- Kommutatorlanadigan tarmoqlari bo'yicha ishlaydigan spamerlar ro'yxati. Bu ro'yxat eskirgan deb hisoblanadi va keyinchalik foydalanish tavsiya etilmaydi.

Yaxshilangan enhdnsbl versiyasi qidiruv natijalariga qarab dasturning hatti-harakatlarini aniqroq aniqlash uchun beshta parametrdan foydalanish imkonini beradi.

Quyidagi yozuvlar qo'llanilgan barcha qora ro'yxatlar maxsus DNS serverlarda saqlanadi:

**IP.blackholes.mail-abuse.org IN A IP**

Sendmail bir nechta qora ro'yxatga murojaat qilish mumkin bo'lgan bir necha marta dnsbl vositasini aniqlash mumkin. Masalan,

```
FEATURE(`dnsbl',`blackhole.mail-abuse.org`,`Rejectet: see www.mail-abuse.org`)dnl
```

```
FEATURE(`dnsbl',`relays.mail-abuse.org`,`Rejectet: see www.mail-abuse.org`)dnl
```

Dnsbl vositasining birinchi argumenti sendmail murojaat qiladigan ro'yxatni aniqlash mumkin. Ikkinchi argument – xato sifatida qaytriladigan satr.

Hozirgi paytda Mail Abuse xizmati pullik servis hisoblanadi. Lekin bepul qora ro'yxatlar ham mavjud. Ularning ro'yxatini quyidagi saytdan ko'rib olish mumkin:

<http://www.decluce.com/junkmail/support/ip4r.htm>



1. Qanday konfiguratsiya parameri sendmail jarayonlari sonini cheklashga imkon beradi?
2. Qanday konfiguratsiya parametri jo'natma xabarining hajmini cheklashga imkon beradi?
3. Xavfsizlik parametrlarida noexpn parametri nima uchun qo'llaniladi?
4. Qadav vositalar sendmaildagi qora ro'yxatni qo'shishda qanday vositalarni qo'llash mumkin?



## 6.13-§. Pochta psevdonim(nomlari)lari

Sendmail pochta nomlarini aniqlashga imkon beradi, foydalanuvchi adresiga qabul qilingan pochta nomi yordamida jo‘natmalarni boshqa pochta qutisiga yoki boshqa adresga yo‘naltirish mumkin.



*Virtusertable vositasini pochta psevdonimlari bilan chalkashtirmaslik kerak. Birinchisi nafaqat foydalanuvchining pochta adresini, balki uning domenini ham hisobga olishga imkon beradi. Pochta psevdonimlari faqat foydalanuvchi nomi bilan ishlaydi.*

Sendmail pochta psevdonimlari bazasini turli xil manbalardan olish imkoniyatini yaratadi:

- psevdonimlar bazasidan tashkil topgan lokal fayllar (jimlik holati bo‘yicha sozlanadi).

- LDAP protokoli bo‘yicha.

- NIS, NIS+ xizmatlaridan.

Biz pochta psevdonimlarining lokal fayli /etc/mail/aliases.db-ni shakllantirishni qarab chiqamiz. Bu ma‘lumotlar bazasidan tashkil topgan binar fayl hisoblanadi. Dastlab matnli fayl aliases hosil qilinib, keyin u aliases.db fayliga o‘tkaziladi.

Pochta psevdonimlari ma‘lumotlar bazasini hosil qilish uchun newaliases dasturidan foydalaniladi.

**Psevdonimlarni aniqlash.** Faraz qilaylik, kompaniya direktoriga tegishli boss nomli qayd yozuvi mavjud bo‘lsin. Direktor qo‘shimcha director nomli e-mail-ni yaratishni talab qilgan.

Yana bitta qo‘shimcha director qayd yozuvini qo‘shish mumkin. Pochta dasturini ikkita pochta qutisi bilan ishlashga sozlash kerak. Bunday ishlash juda noqulay. Shunday hollarda pochta psevdonimidan foydalanish mumkin. /etc/mail/aliases fayliga quyidagi satrni qo‘shish kerak:

**director: boss**

Bu yozuv sendmailga director foydalanuvchisiga kelgan barcha jo‘natmalarni boss foydalanuvchisining pochta qutisiga joylashtirish imkoniyatini yaratadi. Bu vaziyatda director foydalanuvchisi qayd yozuvini hosil qilish shart bo‘lmaydi, sendmail dastlab pochta psevdonimlari ma‘lumotlar bazasi mazmuniga qaraydi va faqat shundan keyin Linux autentifikatsiya tizimiga murojaat qiladi.

Yuboriladigan adres sifatida faqatgina lokal pochta adresini emas, balki e-mail-ni ham ko'rsatish kerak.

**director: boos@samtuit.uz**

Shuningdek, yuboriladigan adreslar ro'yxatini vergul bilan ajratib ko'rsatish mumkin:

**director: boos, spy@samtuit.uz**

Bu vaziyatda maktublar director foydalanuvchisining lokal pochta qutisiga va spy@samtuit.uz e-mailiga yuboriladi.

Adreslar (elektron pochta) ro'yxatini kiritganda, oddiy jo'natmalar ro'yxatini tuzish mumkin.

Odatda root foydalanuvchisiga mo'ljallangan jo'natmalar, tizimning real foydalanuvchisiga yo'naltiriladi.

#### **6.14-§. Elektron pochta adreslari ro'yxatini fayldan yuklash**

Elektron pochta ro'yxatini hosil qilish uchun psedonim nomlaridan keyin darhol barcha adreslarni yozish mumkin. Lekin bu ro'yxat adreslar soni juda ko'p bo'lganda noqulay bo'ladi.

Bunday hollarda barcha e-mail-larni alohida faylning bitta satrida yozish mumkin, faylning o'ziga bog'lanish esa, «include:» direktivasi yordamida amalga oshiriladi.

**list: : include:/fayl/ga/yo`l**

Adreslar fayli NFS bo'yicha bog'langan tarmoq diskida emas, aynan lokal diskda joylashgan bo'lishi shart. Agar fayl tizimlari hard parametri bilan montirovka qilingan va NFS serveri ishdan chiqqan bo'lsa, sendmail dasturi ishi to'xtatiladi.

**Jo'natmani faylga yo'naltirish.** Agar psevdonimlar murojaati qo'yilgan ob'ekt sifatida faylga absolyut yo'l ko'rsatilsa, jo'natma faylning oxiridan qo'shiladi.

**user: /fayl/ga/yo`l**

Pochta xabarlari yuboriladigan fayl DefaultUser parametri yordamida aniqlangan foydalanuvchiga tegishli bo'lishi shart.

**Jo'natmani dasturga yo'naltirish.** Pochta psedonimlari ma'lumotlar bazasining eng qiziqarli jihati jo'natmalarni dasturning standart kirishiga yo'naltirish imkoniyati hisoblanadi.

**manager: |/dastur/ga/yo`l**

Odatda bunday xususiyat yordamida jo'natmalar ro'yxati menedjeri tatbiq qilinadi. Masalan, agar xat Subject maydonida

Subscribe kalit so‘zi bo‘lgan holda uzatilsa, dastur bunday psevdonimni xat yuboruvchilarning e-mailini jo‘natmalar ro‘yxatiga qo‘shishi mumkin. Agar Subject-da Unsubscribe kalit so‘zi bo‘lsa, u holda bu e-mail jo‘natmalar ro‘yxatidan o‘chiriladi.

Dastur sendmail ishlab turgan kompyuter foydalanuvchisi nomidan bajariladi.



*Jo‘natmalarni dasturga yo‘naltirishdan foydalanish – bu tizimni buzishning potentsial imkoniyati hisoblanadi. Shuning uchun ham buni qo‘llashda juda ehtiyot bo‘lish kerak.*

**~/forward fayli.** Aliases fayliga barcha o‘zgartirishlarni faqat super foydalanuvchi amalga oshirishi mumkin. Lekin, tizimda jo‘natmalarni uzatuvchi oddiy foydalanuvchilar ham bor. Ularga jo‘natma uzatiladigan e-mail-ni ~/forward fayliga yozish yetarli bo‘ladi.

Bu faylda aliases faylida qo‘llaniladigan barcha mexanizmlar ishlaydi. Bu fayl uchun yozuvda pochta psevdonimining nomini yozish shart emas.

.forward fayliga misol:

**boss@samtuit.uz**

**spy@samtuit.uz, /home/usr/new\_mail\_box**

Bu fayldagi ma‘lumotlarga asosan, joriy foydalanuvchiga kelgan jo‘natmalar, boss@samtuit.uz va spy@samtuit.uz adreslariga yo‘naltiriladi, shuningdek, /home/usr/new\_mail\_box faylining oxiriga qo‘shiladi.

**Redirect vositasi.** Xodim tashkilotdan ishdan bo‘shatilganda, administrator uning qayd yozuvini o‘chirishi kerak. Bunday vaziyatlarda hodimning eski e-mail-iga yuborilgan jo‘natmalar, xatolik haqida xabarni qaytaradi.

Xodimning talabi asosida ushbu pochta keluvchi jo‘natmalarni yangi e-mailga yo‘naltirish yoki yangi e-mailni saqlab turuvchi xabarni qaytarishga sozlash mumkin.

Xatolik haqidagi xabarni emas yangi e-mail-ni saqlab turuvchi xabarni qaytarishi uchun redirect vositasini qo‘llash kerak bo‘ladi.

**FEATURE (redirect) dn1**

Psevdonimga esa quyidagi satrni yozish kerak:

**user: user@new.com.REDIRECT**

1. Qanday qilib, user foydalanuvchisiga kelgan jo‘natmalarni any foydalanuvchisining pochta qutisiga yo‘naltirish mumkin?



2. Xatlarni tashqi fayldan jo‘natiladigan adreslar ro‘yxatini yuklab olish imkonini beruvchi satrni yozing?

3. Foydalanuvchilar mustaqil ravishda jo‘natmalarni yo‘naltirishi uchun qanday fayldan foydalanishi mumkin?

## 7-BOB. SQUID PROXY SERVERI

Squid proxy serveri quyidagi imkoniyatlarni qo‘llab-quvvatlaydi:

- foydalanuvchining HTTP va FTP protokollari bo‘yicha so‘rovlarini keshlash.
- HTTPS protokoli bilan ishlash.
- proxy serverlar ierarxiasini tashkil etish.
- autentifikatsiya.
- shaffof proxy (transparent proxy).
- resurslarga kirish nazorati (ACL).
- so‘rovlarni qayta yo‘naltirish (redirect),
- kanal kengligini cheklash (delay pools).
- SNMP ni qo‘llab-quvvatlash.
- va bir qancha boshqa imkoniyatlar.

### 7.1-§. Kiruvchi kodlardan dasturni yig‘ish

Aksariyat distributivlarda Squid proksi serveri tayyor paket ko‘rinishida joylashgan bo‘ladi. Slackware Linux-da bunday paket mavjud emas, serverni qo‘lda yig‘ish kerak bo‘ladi.

Qo‘lda yig‘ilgan (yozilgan) serverni kompilyatsiya qilishda ba’zi parametrlar yoqiladi yoki o‘chiriladi. Bu oddiygina configure dasturlari yordamida bajariladi.

Dasturlarning kirish kodlarini olgandan keyin, ular odatda uy katalogiga to‘plab olinadi. Dasturni yig‘ish bo‘yicha qolgan amallar standart holatda bajariladi:

- **configure --help**
- **configure**
- **make**
- **make install**

Birinchi qadamda configure dasturiga beriladigan parametrlarni olamiz

```
./ configure --help
```

Quyida bu parametrlarning ba’zilari keltirilgan:

- prefix=/dir – dastur o‘rnatiladigan katalogni aniqlash.
- enable-delay-pools – kanal kengligiga cheklov o‘rnatishni yoqish.
- enable-arp-acl – kirish ruxsati ro‘yxatiga MAC adreslardan

foydalanish imkoniyatlarini qo‘shish.

`--enable-default-err-language` – xatoliklar haqidagi xabarlarini chiqarishda foydalaniladigan tilni aniqlash.

`--enable-err-language` – proksi server qo‘llab-quvvatlaydigan tillar ro‘yxatini aniqlash.

`--enable-linux-netfilter` – shaffof proksi rejimini (transparent proxy) qo‘llab-quvvatlashni yoqish.

Ikkinchi qadamda, tanlab olingan parametr bilan configure dasturini ishga tushiramiz.

```
./configure --prefix=/opt/squid \  
--enable-delay-pools \  
--enable-arp-acl \  
--enable-default-err-language=Russian-koi8-  
r \  
--enable-err-languages="Russian-koi8-r  
Russian-1251"
```

Shundan so‘ng dastur yig‘iladi:

```
make
```

va o‘rnatiladi:

```
make install
```

Dastur /opt katologiga o‘rnatilganligi sababli, PATH o‘zgaruvchisiga proksi serverning bajariluvchi fayli joylashgan katologni qo‘shish kerak bo‘ladi. Buning uchun /etc/profile.d/squid.sh skriptini quyidagicha mazmunda yozamiz:

```
#!/bin/bash
```

```
export
```

```
PATH=$PATH:/opt/squid/bin:/opt/squid/sbin
```

va uni bajariluvchi faylga aylantiramiz:

```
chmod a+x /etc/profile.d/squid.sh
```

## 7.2-§. Squid.conf konfiguratsiya fayli

Agar siz tayyor paketdan foydalansangiz, u holda proksi serverning konfiguratsiya fayli /etc/squid katologida joylashgan bo‘ladi. Bizning holatda esa, serverni /opt katologidagi kiruvchi kodlardan o‘rnatganimiz uchun konfiguratsiya fayli /opt/squid/etc katologida joylashadi. Bu faylning nomi squid.conf.

Squid.conf faylida administrator o‘zining xohishiga ko‘ra

o'zgartirishi mumkin bo'lgan barcha konfiguratsiya parametrlari tavsiflanadi.

Izoh belgisi - #. Ushbu belgidan keyin yozilgan barcha belgilar satr oxirigacha izoh hisoblanadi. Squid.conf faylidagi izohlar squid parametrlari bo'yicha yetarli darajadagi to'liq hujjat hisoblanadi. Lekin eng to'liq hujjatni <http://www.squid-cache.org> saytidan topish mumkin. Konfiguratsiya parametrlarining soni juda ko'p bo'lganligi uchun, keyingi bo'limlarda faqat eng zarurlarini ko'rib chiqamiz.

Faylda quyidagi keltirilgan satrlarga o'xshash satrlar uchraydi:

```
#Default
```

```
#http_port 3128
```

Xuddi shunday parametrlarning jimlik holati bo'yicha qiymatlar ham keltiriladi. Agar parametрни o'zgartirmoqchi bo'lsak, satr boshidagi izoh belgisini olib tashlashimiz va parametrğa yangi qiymat kiritishimiz kerak bo'ladi.

### **7.3-§. Tarmoqqa ulanishga ta'sir qiluvchi parametrlar**

**http\_port parametri.** Birinchi o'rinda klientlarning proxy server xizmatlariga ulanishida foydalanadigan port nomerini aniqlash kerak bo'ladi. Port nomerini aniqlash uchun http\_port parametri mas'ul.

```
http_port 3128
```

Parametr sifatida IP:port juftligidan foydalanish mumkin. Bunday holatda squid proksi serveri klientlardan so'rovlarni qabul qilish interfeysi va porti aniqlanadi.

Jimlik holati bo'yicha qiymati: 3128.

**https\_port parametri.**

```
[ip:]port cert=certificate.pem
```

```
[key=key.pem] [options...]
```

Bu parametr faqat Sizning WEB serveringizga keshlash rejimida qabul qilinadi. U proxy server qabul qiladigan SSL so'rovlar portini aniqlaydi.

Sertifikatli faylni aniqlovchi cert parametrini ko'rsatish kerak.

Jimlik holati bo'yicha qiymati aniqlanmagan.

**icp\_port parametri.** Squid serveri ICP so'rovlarni qabul qiladigan portni aniqlaydi. Agar proksi server ierarxik strukturaga birlashtiriladigan bo'lsa, bu port komandalarni almashish uchun

qo‘llaniladi. Jimlik holati bo‘yicha qiymati 3130 ga teng.

#### 7.4-§. Proksi serverlar ierarxiasini tashkil etish parametrlari

Squid proksi serveri serverlarni ierarxik strukturaga birlashtirish imkonini beradi. O‘zingizning serveringizni oldindan mavjud serverlar ierarxiasiga ulashingiz kerak bo‘ladi.

Buning uchun qandaydir ixtiyoriy serverga ulanishingiz mumkin, sizda u serverga proksi server klienti sifatida ulanish uchun ruxsat bo‘lishi va ICP portga kirish ochiq bo‘lishi kerak bo‘ladi.

**Cache\_peer parametri.** Bu parametr yordamida Siz ulanishingiz kerak bo‘lgan proksi serverni, shuningdek, ulanishning ba’zi paramerlarini ko‘rsatish mumkin.

Misol uchun:

```
cache_peer parent.foo.net parent 3128 3130
cache_peer sib1.foo.net sibling 3128 3130
cache_peer sib2.foo.net sibling 3128 3130
```

Bu misolda ulanish mumkin bo‘lgan uchta server aniqlangan. Ular bilan o‘zaro muloqot turlari: parent, siblin yoki multicast. Ma’lumotlarni uzatish porti 3128. Komandalarni berish (uzatish) porti 3130.

Jimlik holati bo‘yicha qiymati aniqlanmagan.

**Serverlar o‘rtasida o‘zaro aloqa turlari.** Parent turidagi o‘zaro aloqada, agar bosh (ildiz) server keshida bizga zarur resurslar bo‘lmasa, uning o‘zi resurslarni Internetdan izlaydi. Shundan keyin resurslarni bizning serverga beradi.

Siblin turidagi o‘zaro aloqada, agar ildiz server keshida zarur resurslar bo‘lmasa, uni Internetdan o‘zimiz izlashimiz kerak bo‘ladi.

**cache\_peer\_domain parametri.** Bu parametr yordamida ildiz serverlar o‘rtasida so‘rovlarni aniqroq taqsimlash mumkin.

Misol uchun, WEB serverlarning so‘rovlariga yaxshiroq xizmat ko‘rsatuvchi ru va ua domenlarida joylashgan server mavjud. Ushbu domenlarda joylashgan resurslarga so‘rovlarni ushbu proksi serverlarga yo‘naltirish kerak bo‘ladi. Bunday holda cache\_peer\_domain parametrlarini quyidagicha aniqlash zarur.

```
cache_peer_domain proxy.net.uz .uz .ru
```

Domen nomlarining oldidagi nuqtaga e’tibor bering. Agar nuqta qo‘yilmasa, u holda faqat http://uz va http://ru so‘rovlari proxy.net.ru



serveriga yuboriladi. uz va ru hududidagi qolgan barcha domenlar bu parametr bilan muloqot qila olmaydi. Jimlik holati bo'yicha qiymati aniqlanmagan.

**hierarchy\_stoplist parametri.** So'rovlarda uchraydigan belgilar ketma-ketligini aniqlash parametri. Agar bu belgilar mavjud bo'lsa, u holda bizning server ierarxiyadagi serverga murojaat qilmaydi.

Odatda bu parametr yordamida dinamik generatsiyalanadigan resurslarga murojaat qilganda uchraydigan belgilarni aniqlanadi.

Parametr aniq belgilangan:

**hierarchy\_stoplist cgi-bin ?**

### 7.5-§. Proksi server keshi o'lchamini aniqlash parametrlari

**cache\_mem parametri.** Proxy server keshi uchun ajratilgan tezkor xotira o'lchamini aniqlash parametri. Bu parametr squid jarayonlari qancha xotirani egallashiga ta'sir qilmaydi.



*Tezkor xotirada nafaqat WEB resurslar, balki server keshidagi qidiruv indeksleri ham saqlanadi.*



*O'rtacha proksi serverlar har bir Gbayt diskli keshda 10 Mbayt, unga qo'shimcha 10-20 Mbayt tezkor xotirada saqlanuvchi ob'ektlarga, yana qo'shimcha ravishda cache\_mem parametri tomonidan aniqlangan tezkor xotirani ishlatadi.*

Parametr qiymatlari quyidagicha tanlanadi – agar siz proksi serverning ishlashi uchun X Mbayt xotira ajratishni rejalashtirgan bo'lsangiz, u holda cache\_mem parametrining qiymati X/3 ga teng bo'lishi kerak. Jimlik holati bo'yicha qiymati 8 Mbaytga teng.

**memory\_pools parametri.** Agar memory\_pools parametrining qiymati off-ga teng bo'lsa, u holda proksi server ishlatilmayotgan tezkor xotirani bo'shatadi. Aks holda bari bir server zaruriyat uchun tezkor xotirani tezda taqsimlaydi.

Kam sonli tezkor xotiraga ega bo'lgan serverlarda bu parametrga off qiymatini o'rnatish tavsiya etiladi.

**cache\_swap\_low va cache\_swap\_high parametrlari.** Bu parametrlar proksi server qachon keshni tozalash mexanizmini ishga tushirish kerakligini aniqlaydi.

Proksi server keshi cache\_swap\_high parametri qiymatidan oshib ketmaguncha to'lib boradi. Ushbu qiymatga yetgandan keyin,

keshga ma'lumotlarni joylashtirish to'xtatiladi va tozalash mexanizmi ishga tushiriladi. Keshda saqlanayotgan ma'lumotlar hajmi `cache_swap_low` parametri qiymatiga yaqinlashsa, keshga yana ma'lumotlar joylashadi.

Parametrlarning qiymati foizlarda aniqlanadi. Shuning uchun ham agar keshning o'lchamini o'zgartirsak, kesh maydonidan ratsional foydalanish uchun albatta bu parametrlarning ham qiymatini o'zgartirishimiz kerak bo'ladi.

Jimlik holati bo'yicha qiymatlari:

**`cache_swap_low 90`**

**`cache_swap_high 95`**

**`maximum_object_size` parametri.** Diskdagi keshga joylashtirish mumkin bo'lgan ob'ektning maksimal o'lchamini aniqlash parametri.

Ushbu parametrning aniqlanishiga juda ehtiyot bo'lish kerak.

Ob'ektning o'lchami `cache_swap_high` parametri bilan aniqlangan xotiraning bo'sh qismidan albatta kichik bo'lishi shart. Jimlik holati bo'yicha qiymati 4096 Kbaytga teng.

**`maximum_object_size_in_memory` parametri.** Bu parametr orqali tezkor xotiradagi keshga joylashtirish mumkin bo'lgan ob'ektning maksimal o'lchami aniqlanadi. Jimlik holati bo'yicha qiymati 8 Kbaytga teng.

**`cache_replacement_policy` parametri.** Bu parametr orqali keshdan ma'lumotlarni o'chirish siyosati aniqlanadi. Mumkin bo'lgan qiymatlari:

**`lru, heap GDSF, heap LFUDA va heap LRU.`**

Jimlik holati bo'yicha qiymati esa: `lru`.

## **7.6-§. Log fayllari va keshning joylashish o'rnini aniqlovchi parametrlar**

**`cache_dir` parametri.** Bu parametr diskdagi keshning o'lchami va boshqa parametrlari holatini aniqlaydi. Birinchi parametri – `ufs`, o'zgartirish kerak emas.

Ikkinchi parametri – proksi serverning keshi qaysi katalogda joylashishini aniqlaydi.

Uchinchi parametri – diskning kesh uchun ajratilgan Mbaytlardagi o'lchami.

To'rtinchi parametri – bu birinchi darajali kataloglar soni.

Beshinchi parametri – bu ikkinchi darajali katologlar soni.

Yana bir qancha muhim bo‘lmagan parametrlarini aniqlash mumkin. Nima uchun diskdagi keshda shuncha sondagi katologlarni hosil qilish kerak? Fayllarga tezkorlik bilan kirish uchun. Agar ob’ektlar katologlar bo‘yicha taqsimlangan bo‘lsa, hamma fayllar bitta katologda joylashgan holatga nisbatan ularga kirish yanada tezroq amalga oshiriladi. Shuning uchun ham keshning hajmi qancha katta bo‘lsa, shuncha ko‘p sondagi katologlarni hosil qilish zarur.

Jimlik holati bo‘yicha qiymati distributivga bog‘liq.

**cache\_access\_log parametri.** Parametr squid proksi serveri kim va qachon proksi serverga murojaat qilganligi, qanday ob’ektlar so‘ralganligi, shuningdek, u so‘ralgan ob’ektni olgan yoki olmaganligi haqidagi ma’lumotlarni joylashtiradigan fayl nomini aniqlaydi.

Bu faylning hajmi juda tez oshib boradi. Har bir so‘rovda faylga 65 bayt atrofidagi bitta satr yoziladi.

Jimlik holati bo‘yicha uning qiymati distributivga bog‘liq.

**cache\_log parametri.** Bu parametr squid proksi serverini sozlash ma’lumotlari joylashadigan fayl nomini aniqlaydi.

Faylda qanchalik darajada to‘liq ma’lumotlar joylashganligini debug\_options parametri orqali aniqlanadi.

Jimlik holati bo‘yicha uning qiymati distributivga bog‘liq.

**cache\_store\_log.** Bu parametr keshga qanday ob’ektlar qaerdan kelib tushganligi haqidagi ma’lumotlar joylashgan faylni aniqlaydi. Qanday ob’ektlar keshdan o‘chirilganligi va ular qancha vaqt keshda saqlanganligi.

Fayl hajmi juda tez oshib boradi, lekin undan foydalanishni faylga yo‘l ko‘rsatish o‘rniga none kalit so‘zini qo‘llash orqali o‘chirib qo‘yish mumkin.

Jimlik holati bo‘yicha uning qiymati distributivga bog‘liq.

**debug\_options parametri.** Bu parametr orqali cache.log fayliga qanchalik batafsil ma’lumot joylashganligi aniqlanadi. Jimlik holati bo‘yicha qiymati: debug\_options ALL,1.

## 7.7-§. Qo‘shimcha dasturlarning ishlashiga ta’sir etuvchi parametrlar

**ftp\_user parametri.** ftp serverlariga anonim foydalanish uchun parol sifatida qo‘yiladigan e-mail adresni aniqlaydi.

Haqiqiy e-mail ko‘rsatish maqsadga muvofiq bo‘ladi, chunki

ba'zi ftp serverlar e-mailning mavjudligini aniqlashga harakat qiladi. Natija noaniq bo'lganda serverga kirishni taqiqlab qo'yadi. Jimlik holati bo'yicha qiymati: ftp\_user Squid@

**ftp\_passive parametri.** Bu parametr ftp serverlar bilan ishlash vaqtida ma'lumotlarni uzatish rejimini faol yoki passiv ekanligini aniqlaydi.

Jimlik holati bo'yicha qiymati: ftp\_passive on

Shuningdek, so'rovlarni qayta yo'naltirish uchun dasturlar va foydalanuvchilarni autentifikatsiya qilish uchun dasturlarni aniqlashga mo'ljallangan parametrlar ham mavjud. Bu parametrlarni alohida bo'limlarda ko'rib chiqamiz.

### 7.8-§. Kirishni nazorat qilish ro'yxati (ACL).

Squidda resurslarga kirishni cheklash ikki bosqichda amalga oshiriladi:

- so'rov shartlarini aniqlash.
- birinchi qadamda aniqlangan shartni qanoatlantirish yoki unga taqiq qo'yish.

So'rov shartlarini aniqlash uchun acl parametri qo'llaniladi.

```
acl aclname acltype string1 ...
```

```
acl aclname acltype "file" ...
```

Aclname – siz kiritadigan acl ning nomi. Bu nom ingliz harflaridan tashkil topgan bo'lishi va unda probel ishlatilmasligi shart.

Acltype – acl-ning turi, ajratilgan so'z.

Keyin acl-ning turiga bog'liq bo'lgan qo'shimcha parametrlarni ko'rsatish zarur. Agar parametrlar juda ko'p bo'lsa, ularni alohida satrlarda tashqi faylga joylashtirish mumkin. Bu faylga yo'l parametr bilan qo'shtirnoq ichida ko'rsatiladi.

**Src turidagi acl.** Ushbu acl so'rov proksi serverga IP adres bilan kelgan vaziyatlarni aniqlaydi. Parametr sifatida quyidagilarni keltirish mumkin:

- IP adres. Misol uchun, 172.20.21.1/32.
- tarmoq IP adresi. Misol uchun, 172.20.21.0/24.
- IP adreslar chegarasi. Misol uchun, 172.20.21.1 – 172.20.21.12/32

Ixtiyoriy IP adresni yozishda tarmoq osti niqobini albatta ko'rsatish talab etiladi. Kompyuterlar IP adresi ko'rsatilgan vaziyatlarda 255.255.255.255 niqobi yoki /32 formatidagi CIDR

qo‘llaniladi.

Acl-ni aniqlashda bir nechta parametrlarni probel bilan ajratilgan holda ko‘rsatish mumkin. Ular mantiqiy YOKI bilan birlashtiriladi.

Acl-ga misollar.

```
acl myhost src 172.20.21.2/32
```

```
acl mynet src 172.20.21.0/24
```

```
acl fr src 172.20.21.1-172.20.21.12/32
```

```
acl any src 172.20.21.21/32 172.20.21.54/32
```

Birinchi misolda, 172.20.21.2 IP adresli kompyuterdan kiruvchi so‘rovlarni tavsiflovchi acl myhost aniqlangan.

Ikkinchi misolda 172.20.21.0/24 tarmog‘idan keladigan ixtiyoriy so‘rovni aniqlovchi acl mynet aniqlangan.

Uchinchi misolda, IP adreslar qanday aniqlanishi ko‘rsatilgan.

To‘rtinchi misolda 172.20.21.21 IP adresli yoki 172.20.21.54 IP adresli kompyuterlardan so‘rov kelgan holatining aniqlashi ko‘rsatilgan.

**Dst turidagi acl.** Ushbu acl ko‘rsatilgan IP adresdagi kompyuterda joylashgan resursga foydalanuvchi tomonidan so‘rov bergan vaziyatni aniqlaydi.

Bu turdagi acl juda kamdan kam vaziyatlarda qo‘llaniladi, ya’ni qiziqtirayotgan WEB sayt o‘z IP adresini o‘zgartirishi mumkin va siz kiritgan cheklovlar ishlashni to‘xtatadi.

**Dstdomain turida acl.** Bu turdagi acl foydalanuvchi resursga nomini ko‘rsatgan holda so‘rov yuborgan vaziyatni tavsiflash uchun xizmat qiladi.

Parametr sifatida resursning to‘liq yoki to‘liq nomini ko‘rsatish mumkin.

Misol uchun:

www.samtuit.uz – www.samtuit.uz saytiga so‘rovni aniqlaydi

.samtuit.uz – nomi .samtuit.uz bilan tugallanuvchi ixtiyoriy kompyuterga so‘rovlarni aniqlaydi.

Acl-ga misollar:

```
acl ilhom dstdomain .samtuit.uz
```

```
acl ramblerdstdomain www Rambler.ru
```

```
acl ya dstdomain .ya.ru .yandex.ru
```

Misoldagi birinchi acl klient tomonidan kompyuter nomi .samtuit.uz bilan tugallanadigan ixtiyoriy kompyuterga so‘rovni

yuborgan vaqtdagi vaziyatni aniqlaydi.

Misoldagi ikkinchi acl klient www.rambler.ru kompyuteriga so'rovni yuborgan vaziyatni tavsiflaydi. Agar foydalanuvchidan rambler.ru domenining boshqa kompyuteri (masalan, mail.rambler.ru) ga so'rov yuborilsa, bu so'rov ushbu misoldagi Acl-ga mos kelmaydi.

Misoldagi oxirgi acl - nomlari .ua.ru yoki .yandex.ru bilan tugallanadigan barcha kompyuterlar uchun klientlardan yuboriladigan so'rovlarni izlash vaziyati ta'riflangan.

**dstdom\_regex turidagi acl.** Bu acl-ning parametri sifatida murojaat qilgan klientning kompyuter nomini aniqlovchi muntazam ifodadan foydalanish mumkin.

Harflarning holatini (katta yoki kichik) e'tiborga olmaslik uchun muntazam ifodani yozishdan oldin -i parametridan foydalanish kerak.

Acl-ga misol:

```
acl anymail dstdom_regex -i mail
```

Bu acl FQDN-dagi uning nomida mail so'zi uchraydigan foydalanuvchi kompyuteriga murojaat qilgan vaziyatni aniqlaydi.

**urlpath\_regex turidagi acl.** Ushbu acl-ni klientning so'rovida uchraydigan so'z yoki so'zning qismini aniqlash uchun qo'llash mumkin. Faqat so'ralgan resurs va resurs nomiga yo'l qaraladi.

Acl-ga misol:

```
acl media urlpath_regex -i \.mpg$ \.avi$ \.mp3$
```

Ushbu acl foydalanuvchi .mpg, .avi, .mp3 kengaytmalariga ega bo'lgan fayllarni olishga uringan vaziyatni tavsiflaydi.

**Time turidagi acl.** Bu acl yordamida klientdan qaysi vaqtda so'rov kelganligini aniqlash mumkin. Unga parametr sifatida vaqt oralig'ini ko'rsatish kerak bo'ladi.

Acl-ga misol:

```
acl wt time 10:00-17:00
```

**Maxconn turidagi acl.** Bu acl bitta klientning ulanishlar soni acl-da ko'rsatilgan sondan oshib ketmaganligini tekshirish imkonini beradi.

Acl-ga misol:

```
acl connections maxconn 4
```

Yuqorida sanab o'tilganlardan tashqari acl-ning turlari juda ko'p. Ularning ba'zilari bilan keyinchalik tanishib chiqamiz.

## 7.9-§. Kirishni cheklash, http\_access parametri

Acl parametrlari yordamida so'rovlarning turli xil shartlarini ro'yxatga olgandan keyin, ushbu shartlarni qanoatlantirish yoki taqiqlash kerak bo'ladi. Buni http\_access parametrlari bilan amalga oshirish mumkin.

Faraz qilamiz quyidagi acl-lar tavsiflangan bo'lsin:

```
acl myhost src 172.20.21.2/32
acl mynet src 172.20.21.0/24
acl wt time 10:00-17:00
acl rambler dstdomain .rambler.ru
```

Kompyuteringizdagi resurslarga kirishga ruxsat berishingiz kerak. Resurslar va internetga 172.20.21.0/24 tarmog'ida ishlaydigan kompyuterlarga ruxsat beriladi, lekin faqat ish vaqtida. Shuningdek, ularga kirish nomi .rambler.ru bilan tugallanuvchi barcha kompyuterlar uchun taqiqlangan.

Buning uchun quyida sanab o'tilganlarni qo'shish kerak:

```
http_access allow myhost
http_access deny rambler
http_access allow mynet wt
http_access deny all
```



*Shuni alohida e'tiborga olish kerakki, http\_access parametrlarini tavsiflash tartibi qoidalarni qarab chiqishda hisobga olinadi. Agar qoida ishlab turgan bo'lsa, ro'yxatdagi keyingi qoida hisobga olinmaydi. Acl tavsiflash tartibi ixtiyoriy bo'lishi mumkin, asosiysi, ular konfiguratsiya faylida qo'llanilmasdan oldin aniqlangan bo'lishi kerak.*

Keltirilgan misolda dastlab acl myhost yordamida aniqlangan kompyuterlarning ixtiyoriy resurslariga kirishga ruxsat berilgan.

Keyin esa acl rambler yordamida tavsiflangan resurslarga kirish taqiqlangan.

Uchinchi satrda resurslarga kirishga ruxsat berilgan, lekin ko'rsatilgan acl-dagi ikkala: wt va time shartlari ishlatilgan. http\_access parametrlari sifatida bir nechta acl-larni probellar orqali ko'rsatish mumkin. Bu holatda ular mantiqiy VA yordamida

birlashtiriladi.

Va nihoyat, barcha qolganlarga kirish taqiqlanadi. Acl all jimlik holati bo'yicha tavsiflanadi va so'rovlarning ixtiyoriy manbasini bildiradi.

### 7.10-§. Boshqaruv (administrativ) parametrlari

**cache\_mgr parametri.** Bu parametr proksi server administratorining e-mail-ni aniqlaydi. Bu e-mail xatoliklar sahifasida chiqadi.

**cache\_effective\_user va cache\_effective\_group parametrlari.** Bu parametrlar proksi serverda ishlash huquqiga ega bo'lgan foydalanuvchi va guruhlarini aniqlaydi.



*Proxy server keshida joylashgan katalog va log fayllari ushbu foydalanuvchilar uchun yozish rejimida ochiq bo'lishi shart.*

Jimlik holati bo'yicha qiymati distributivga bog'liq.

#### **Proxy server bilan ishlash, uni ishga tushirish va to'xtatish.**

Squid proxy serverini ishga tushirish va to'xtatishni boshqarish ishga tushirish skriptlari yordamida, shuningdek, dasturlarni shaffof chaqirish yo'li bilan bajarilishi mumkin.

**Proxy server keshini tayinlash.** O'rnatish va squid.conf konfiguratsiya faylini birinchi o'zgartirishdan keyin diskda proksi serverning keshini tayinlash zarur bo'ladi. Buni quyidagi komandalar satri orqali bajarish mumkin:

**squid -z -D -F**

-z parametri keshni tayinlashni ishga tushiradi, ya'ni soni cache\_dir parametrlari bilan aniqlangan kesh katologlari ishlashi uchun zarur bo'lgan barchasini hosil qiladi.

-D parametri squid-ning ishga tushishida DNS-ning ish unumdorligini tekshirishni o'chiradi. Proksi serverni tekshirishda ma'lum saytlarning IP adreslarini olishga harakat qiladi: www.microsoft.com, www.netscape.com va boshq. -D parametridan proksi server o'rnatilgan kompyuter Internet yoki DNS serverga kirish huquqiga ega bo'lmagan vaqtda foydalanish tavsiya etiladi.

-F parametri – proksi serverga keshni hosil qilish vaqtidagi so'rovlarga ishlov bermaganligini bildiradi.

**Squid proxy serverini ishga tushirish va to'xtatish.** Proksi



serverni ishga tushirish va to'xtatish quyidagi komandalar satri yordamida amalga oshiriladi:

```
squid
```

Agar zarurat bo'lsa, ishga tushirishda `-D` parametrini qo'shish mumkin.

Proksi serverni to'xtatish 15 signalini yuborish orqali amalga oshiriladi:

```
killall squid
```

yoki quyidagicha:

```
squid -k shutdown
```

**Proksi serverni kompyuter ishga tushirilganda yuklash.**

Slackware Linux-da proksi server ishlashi uchun ishga tushirish skripti qo'yilmagan. Shuning uchun ham yuklanishni `/etc/rc.d/rc.local` faylida tavsiflash zarur bo'ladi. Bu faylga quyidagi satrlarni qo'shish zarur:

```
if [ -f /opt/squid/var/logs/squid.pid ];  
then
```

```
rm /opt/squid/var/logs/squid.pid
```

```
fi
```

```
/opt/squid/din/squid
```

`/opt/squid/var/logs/squid.pid` faylida squid dasturining PID-i saqlanadi. Agar dastur noto'g'ri o'chirilgan bo'lsa, u holda u ushbu faylni o'chirmaydi. Proksi server ishga tushishida ushbu faylning mavjudligi tekshiriladi va u mavjud bo'lsa, proksi server yuklanmaydi. Shuning uchun ham biz dastlab ushbu faylni o'chiramiz. Va shundan keyin proksi serverni ishga tushiramiz.

Bu fayl qanday nomlanishi va qaerda saqlanishi, `pid_filename` parametriga bog'liq.

### **7.11-§. Proxy serverni boshqarish**

Agar proksi server ishlayotgan vaqtda konfiguratsiya fayliga o'zgartirishlar kiritilgan bo'lsa, serverni konfiguratsiya faylini qayta o'qishga majburlash kerak:

```
squid -k reconfigure
```

**Squid proxy serverining log fayllarini almashtirish.** Squid proksi serveri log fayllarni boshqarish uchun Syslog tizimining xizmatlaridan foydalanmaydi. Bu log fayllariga joylashadigan ma'lumotlarning miqdori ko'p ekanligini bildiradi. Shuning uchun ham log fayllarni almashtirib turishni proksi serverning o'zi orqali

amalga oshirish kerak bo‘ladi:

```
squid -k rotate
```

### 7.12-§. Autentifikatsiyani sozlash

Squid proksi serveri foydalanuvchilarni autentifikatsiya qilish imkonini beradi, chunki squidning o‘zi autentifikatsiyaning to‘g‘riligini tekshirmaydi, u faqat foydalanuvchining login va parolini so‘raydi. Foydalanuvchidan olingan ma’lumotlarning to‘g‘riligini tekshirish uchun tashqi dasturlardan foydalanadi.

Autentifikatsiyani sozlash uchun `auth_param` parametri qo‘llaniladi:

```
auth_param sxema parametr [options]
```

“sxema” parametri autentifikatsiya sxemasini aniqlaydi: `ntlm`, `digest` yoki `basic`.

Internet Explorer-da agar ro‘yxatda birinchi bo‘lib `basic` sxemasi aniqlangan bo‘lsa, proxy server boshqa sxemalarni qo‘llashi yoki qo‘llamasligidan qat’iy nazar u faqat ushbu `basic` sxemani qo‘llaydi. Shuning uchun ham sxemalarni tavsiflash tartibi ahamiyatga ega.



*Proksi serverning yangi sxemani qo‘llashini ishga tushirish yoki o‘chirishdan keyin uni to‘xtatish va yuklash tavsiya etiladi.*

Biz autentifikatsiyaning eng oddiy `basic` variantini `ncsa_auth` dasturidan foydalanib hosil qilishni qarab chiqamiz.

Birinchi o‘rinda proksi server bilan birga qo‘yilgan kiruvchi kodlardan dasturni yig‘ish kerak bo‘ladi. Bu kiruvchi kodlar odatda `helpers/basic_auth/NCSA` katologida joylashgan bo‘ladi. Ushbu katologga o‘tish va `make` dasturini ishga tushirish zarur.

Dasturni olgandan keyin uning nusxasini `/opt/squid/sbin` katologiga o‘tkazish kerak.

`ncsa_auth` dasturining ishlashi uchun foydalanuvchilar va ularning parollari yozilgan faylni hosil qilish zarur bo‘ladi. Ushbu faylda keltirilgan foydalanuvchilar Linux foydalanuvchilari bilan mos kelmasligi mumkin.

`/opt/squid/etc/passwd` faylini hosil qilish va uni `htpasswd` dasturi yordamida tahrirlash kerak bo‘ladi. Bu dastur Apache WEB serveri bilan birgalikda qo‘yilgan.

```
htpasswd -c /opt/squid/etc/passwd user
```

```
-s parametrini faqat /opt/squid/etc/passwd fayli mavjud
```

bo‘lmagan hollarda ko‘rsatish kerak bo‘ladi.

User - bu ushbu faylga qo‘shadigan foydalanuvchi nomi yoki parolini o‘zgartirish zarur bo‘lgan foydalanuvchi nomi.



*/opt/squid/etc/passwd fayliga kirish cheklangan bo‘lishi shart.*

Passwd faylini hosil qilgandan keyin, proksi serverning konfiguratsiya fayliga o‘zgartirish kiritish zarur.

```
auth_param          basic          program
/opt/squid/sbin/ncsa_auth /opt/squid/etc/passwd
```



*Yuqorida keltirilgan yozuv albatta bitta satrda yozilishi shart*

```
auth_param basic children 5
auth_param basic realm Squid proxy-caching
web server
auth_param basic credentialsttl 2 hours
```

Birinchi satrda basic sxemasidan foydalanayotgan holatlarda ishlatiladigan dastur aniqlangan. Dasturga to‘liq yo‘l va shuningdek uning ishlashi uchun zarur bo‘lgan parametrlar ko‘rsatilgan.

Ikkinchi satrda bir vaqtda ishga tushirilgan autentifikatsiya dasturlarining soni aniqlangan. Nechta dastur ishlayotganligi autentifikatsiya jarayonini tezlashtirish uchun zarur bo‘ladi.

Uchinchi satrda klientga beriladigan nom aniqlangan.

To‘rtinchi satrda squid login va parollarni autentifikatsiya dasturidan tasdiqlash talab etilmasligi uchun qancha vaqt saqlashi aniqlangan.

**Autentifikatsiyani qo‘shish (yoqish).** Autentifikatsiyaning asosiy sozlamalarini bajarib bo‘lgandan keyin, uni qo‘shish zarur bo‘ladi. Bu acl va http\_access parametrlari yordamida bajariladi.

Dastlab, proxy\_auth turidagi acl-ni aniqlash kerak. Misol uchun:

```
acl passwd proxy_auth REQUIRED
```

```
acl any_auth proxy_auth anna bell thom
```

Birinchi satrda barcha foydalanuvchilar uchun autentifikatsiya so‘rovi aniqlangan. Ikkinchi foydalanuvchilar aniq ko‘rsatilgan.

Resurslarga parol yordamida kirishni http\_access parametri yordamida cheklash mumkin. Misol uchun, internetdagi media fayllarga kirishni cheklash zarur bo‘lsa, u holda ikkita acl va bitta

http\_access-ni aniqlash kerak bo‘ladi.

```
acl auth proxy_auth REQUIRED
```

```
acl media urlpath_regex -I \.mpg$ \.avi$ \.mp3$
```

```
http_access allow media auth
```

```
http_access deny media
```

http\_access parametrtdagi acl-ning tartibiga e’tibor bering. Autentifikatsiyani tavsiflovchi acl ro‘yxatning oxirida turishi kerak. Bu agar foydalanuvchi media resurslarni so‘ramayotgan bo‘lsa, http\_access ko‘rsatmalarini tahlil qilishda autentifikatsiya mexanizmini chaqirmaslik uchun qilingan. Ya’ni, dastlab birinchi acl tekshiriladi va agar shartlar mos kelsa, u holda faqat ikkinchi acl tekshirishi amalga oshiriladi.

### 7.13-§. Kanal kengligini cheklash.

Squid proksi serverida kanal kengligini cheklash imkoniyati o‘rnatilgan. Cheklovlar acl-ga qo‘shib qo‘yilgan, ya’ni vaziyatga bog‘langan. Misol uchun, aniq kompyuter yoki foydalanuvchi uchun ajratilgan hamma trafikni cheklab qo‘yish mumkin. Bundanda murakkab vaziyatlarni ham aniqlash mumkin, misol uchun, agar foydalanuvchi aniq bir turdagi fayllarni tarmoqdan ko‘chirib olmoqchi bo‘lsa yoki ftp protokolidan foydalanmoqchi bo‘lsa, unga cheklovlar o‘rnatish mumkin.

Cheklovlarni o‘rnatish uchun sig‘imni ushlab qoluvchi (delay pools) deb nomlangan parametrlar guruhi qo‘llaniladi.

Birinchi navbatda delay\_pools parametri yordamida sig‘im miqdori aniqlab olinadi. Misol uchun, ikkita sig‘imni aniqlash uchun quyidagicha yozish zarur bo‘ladi:

```
delay_pools 2
```

Sig‘im uchta sinfga kiradi:

Birinchi sinf, sig‘imga ulangan barcha acl trafiklarini cheklash uchun mo‘ljallangan.

Ikkinchi sinf, C sinfidagi tarmoqlar va har bir acl uchun alohida cheklovlarni aniqlaydi.

Uchinchi sinf, V sinfidagi tarmoq uchun cheklovlarni aniqlaydi, keyin S sinfidagi qismtarmoq uchun alohida cheklovlarni va yana har bir foydalanuvchi uchun cheklovlarni aniqlaydi.

Sig‘im qaysi sinfga tegishli ekanligini aniqlash uchun delay\_class parametri qo‘llaniladi:

```
delay_class 1 1
```

```
delay_class 2 1
```

Birinchi son sig'im nomerini, ikkinchisi esa uning sinfini aniqlaydi.

Sig'implarning sinfi va sonini aniqlagandan keyin, cheklash parametrini berish kerak. Bu `delay_parameters` parametri yordamida bajariladi.

`delay_parameters` parametrining variant (optsiya)lari soni sig'im sinfiga bog'liq:

- birinchi sinf sig'imi – bitta parametr.
- ikkinchi sinf sig'imi – ikkita parametr.
- uchinchi sinf sig'imi – uchta parametr.

Cheklovlarni aniqlashda bitta parametrga ikkita qiymat: tezlik va sig'im hajmi ko'rsatilishi zarur. Misol uchun, sekundiga 64 Kilobaytdan 800 bitgacha o'lchamdagi ob'ektlar uchun ko'chirib olish tezligini cheklash zarur bo'lsin. Buning uchun quyidagi qiymatlar juftligi qo'llanilishi kerak: 800/64000.

Birinchi sinfdagi sig'im uchun cheklovni aniqlash holatida `delay_parameters` parametrini quyidagi ko'rinishda yozsa bo'ladi:

```
delay_parameters 1 800/64000
```

Bu yerda 1 – sig'im nomeri.

Agar 800/64000 ning o'rniga 800/800 yozilsa, ko'chirib olish tezligi sekundiga 800 bit bo'ladi.

-1 soni hech qanday cheklov yo'qligini bildiradi.

Agar 2-sinf sig'implari uchun parametrlarni aniqlansa, ikki juft qiymatlarni ko'rsatish kerak: tarmoq uchun va har bir kompyuter uchun. Masalan:

```
delay_parameters 1 64000/64000 4000/4000
```

Barcha tarmoqlar uchun umumiy trafikni sekundiga 64 Kilobayt, har bir kompyuter uchun sekundiga 4 Kilobayt cheklash o'rnatildi.

Sig'imning barcha parametrlari aniqlanganda, unga qaysi acl ulanganligini tavsiflash kerak bo'ladi. Bu `delay_access` parametri yordamida bajariladi. Misol uchun, bizga multimediali fayllarni ko'chirib olish uchun sekundiga 400 Kilobaytgacha tezlikni cheklash kerak bo'lsin.

Dastlab mul'timedia fayllarining barcha turlarini tavsiflovchi acl-ni aniqlab olamiz:

```
acl media urlpath_regex -i \.mpg$ \.avi$  
\.mp3$
```

Keyin ushlab qolinadigan sig'imni aniqlaymiz:

```
delay_pools 1
delay_class 1 1
delay_parameters 1 400/400
delay_access 1 allow media
delay_access 1 deny all
```

delay\_access parametrini aniqlashda albatta acl-ga ulanadigan sig'im nomerini ko'rsatiladi.



1. Squid proxy serverining asosiy imkoniyatlarini sanab bering?
2. cache\_swap\_low va cache\_swap\_high parametrlari nimaga ta'sir qiladi?
3. Acl parametrini qanday tartibda aniqlash kerak?
4. Proksi serverning ishiga http\_access parametrining aniqlanish tartibi ta'sir qiladimi?
5. Squid proksi serverini qanday qilib o'zining konfiguratsiya faylini qayta o'qishiga majburlash mumkin?

#### 7.14-§. Squid proxy serverining log fayllari bilan ishlash

Squid proksi server ishlashida turli xil log fayllarni ishlatadi. Ularning ichidan /var/log/squid/access.log faylini ajratib olish mumkin. Bu faylga kim va qanday resurslarni olishga harakat qilgan, resurslarga kirish ruxsatini olganmi yoki yo'qligi, va proksi serverdan foydalanish bo'yicha boshqa ma'lumotlar tushadi.

```
1081501287.175 64 127.0.0.1 TCP_MISS/200 1216
GET
http://top.list.ru/counter& -
DIRECT/194.67.45.100 image/gif
```

Foydalanuvchining resursga har bir so'rovida bu faylga uzunligi 60-70 bayt atrofidagi (so'rovga bog'liq) bitta satr qo'shiladi. Bundan kelib chiqadiki, bu faylning hajmi juda tez kattalashib boradi.

Faylning hajmi oshishini cheklash (rotatsiyalash) uchun squid-ga o'rnatilgan imkoniyatlardan foydalanish mumkin.

```
squid -k rotate
```

Lekin dastlab squid.conf konfiguratsiya faylida logfile\_rotate

parametrini aniqlash va saqlanadigan fayllar sonini ko'rsatish zarur. Agar buni amalga oshirmasak, sukut bo'yicha qiymati – 0 bo'ladi va fayllarni almashtirish taqiqlanadi.

Almashtirishdan keyin yangi fayl paydo bo'ladi, misol uchun access.log.o.

Almashtirish jarayoni tez-tez bajarilishi access.log faylining hajmi oshishi tezligiga bog'liq.

Proksi server fayllarini almashtirish uchun logrotate dasturidan foydalanish tavsiya etilmaydi. Eng yaxshi variant CRON-ni qo'llash hisoblanadi.

/etc/cron.daily katologida quyidagi mazmunda skript hosil qilish zarur:

```
#!/bin/bash
```

```
squid -k rotate
```

va uni bajariluvchi qilishni esdan chiqarmang. Endi kuniga bir marta serverning log fayllarini almashtirish bajariladi.



1. Qaysi parametr yordamida klientlar va ularning proksi serverga so'rovlari haqidagi ma'lumotlar yoziladigan fayl qaerda joylashganligi va qanday nomlanishi aniqlanadi?

2. Qanday parametr log fayllarni almashtirishni qo'shadi?

3. Proksi serverning log fayllarini almashtirishni tez-tez bajarish qanday bajariladi?

## 8-BOB. LINUX-DA PAKETLARNI FILTRLASH (FIREWALL)

### 8.1-§. Lunux-da paketlarni filtrlash (firewall)

Lunux yadrosida tarmoq paketlarini filtrlash imkoniyati o‘rnatilgan. 2.4.x va 2.6.x versilarining yadrosida bu dasturiy ta‘minot netfilter deb ataladi.

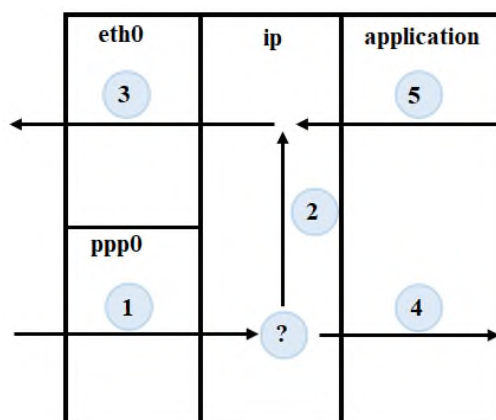
Paketlarni filtrlashni boshqarish uchun administratorlar filtrlash parametrlarini sozlashda foydalanadigan maxsus dasturiy ta‘minot qo‘llanishi kerak. Yadro versiyasiga bog‘liq holda boshqarish uchun turli xil dasturlar qo‘llaniladi:

- 2.0.x - **ipfwadm**
- 2.2.x - **ipchains**
- 2.4.x va 2.6.x - **iptables**

Paketlarni filtrlashni boshqarishning uchta turli xil dasturlarning mavjudligiga qaramasdan, yadroning zamonaviy versiyalarida ulardan ixtiyoriy bittasidan foydalanish mumkin, ya‘ni, Linux ularning barchasini qo‘llab-quvvatlaydi. Lekin Linux-ning zamonaviy yadrolarida paketlarni filtrlashning barcha imkoniyatlarini qo‘llovchi iptables dasturidan foydalanish tavsiya etiladi.

Bu bobda iptables dasturi yordamida paketlarni filtrlashni tashkil etish va NAT akslantirishni hosil qilish o‘rganiladi.

**Zanjirlar.** Ipchains dasturining paydo bo‘lishi bilan zanjirlar tushunchasi kiritilgan. Zanjir – bu filtrlash qoidalarini ulash mumkin bo‘lgan paketlar o‘tadigan ba‘zi mantiqiy yo‘llar qismi.



Filtrlash qoidalarini tashkil etishni tushunib olishni yengillashtirish uchun faraz qilamizki, barcha paketlar ppp0 (1)



interfeysi orqali tushadi, chiqish esa etho (3) interfeysi orqali bajarilsin. Paket IP darajasiga tushgandan keyin uni marshrutlash haqida qaror qabul qilinadi. Marshrutlash haqidagi qarordan keyin paket ushbu kompyuterda ishlaydigan ilovalarga (4) beriladi yoki boshqa tarmoq interfeysi orqali chiqib (2,3) ketadi. Shuningdek, paketlar kompyuterda ishlayotgan ilovalarni yuborilishi ham mumkin (5,3).

Rasmda tasvirlangan strelkalar filtrlash qoidalariga ulanishi mumkin bo'lgan zanjirlar hisoblanadi.

Nomi	Nomeri
PREROUTING	1
POSTROUTING	3
INPUT	4
FORWARD	2
OUTPUT	5

Zanjirlar nomi zahiralangan va ular bosh harflar bilan yozilgan.

Ipchains-da zanjir nomlari bilan va ular orqali paketlarning o'tishida muammo mavjud. Agar tranzitlar bizning kompyuter orqali o'tadi deb faraz qilsak, u ketma-ket INPUT (1), FORWARD (2) va OUTPUT (3) zanjirlari orqali o'tadi. Bu holda paket har bir zanjirda joylashgan qoidalar bilan tekshiriladi, bu paketning o'tish tezligini susaytiradi.

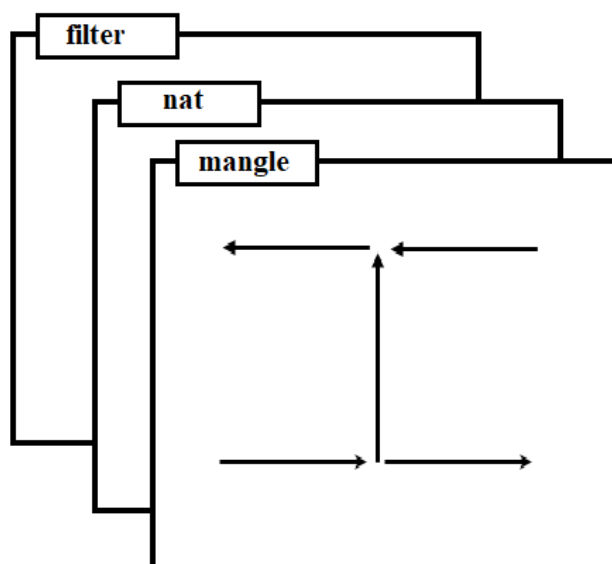
Iptables bu kamchiliklardan holi bo'lib, tranzit paketlar faqat FORWARD (2) zanjiri qoidalari orqali o'tadi. Kompyuterdagi ilovalar uchun mo'ljallangan paketlar faqat INPUT (4) zanjiri qoidalari orqali o'tadi. Kompyuterning ilovalaridan chiquvchi paketlar OUTPUT (5) zanjiri qoidalarida qayta ishlanadi.

Ipchains va iptables-da foydalanuvchilar o'zlarining shaxsiy zanjirini aniqlashlari mumkin.

**Jadvallar.** Iptables-da zanjirlardan tashqari uchta jadval qo'llaniladi:

- filter
- nat
- mapgle

Har bir jadval zanjirlarning individual to'plamiga ega. Oldingi bo'limda qaralgan misolda filter jadvalining zanjirlari tavsiflandi.



**Filter jadvali.** Jadval paketlarni filtrlash qoidalarini aniqlash uchun mo‘ljallangan.

Jadvalda uchta zanjir mavjud:

- INPUT (4)
- FORWARD (2)
- OUTPUT (5)

**Nat jadvali.** Jadval NAT (Network Address Translations) qoidalarini tavsiflash uchun mo‘ljallangan.

Jadvalda uchta zanjir mavjud:

- PREROUTING (1)
- POSTROUTING (3)
- OUTPUT (5)

**Mangle jadvali.** Jadval paketlarning sarlavhasiga o‘zgartirish kiritish uchun mo‘ljallangan.

Jadvalda beshta zanjir mavjud:

- PREROUTING (1)
- POSTROUTING (3)
- INPUT (4)
- FORWARD (2)
- OUTPUT (5)

## 8.2-§. Zanjirlar va jadvallardan o‘tish tartibi

Netfilter bir nechta jadval va zanjirlarni qo‘llashiga qaramasdan, paket parallel ravishda barcha jadvallar va zanjirlardan o‘tmasligi mumkin. U ular orqali ketma-ket o‘tadi.

Tranzit paketlarning o'tish tartibi.

<b>Jadvallar</b>	<b>Zanjirlar</b>
mangle	PREROUTING
nat	PREROUTING
mangle	FORWARD
filter	FORWARD
rr.angle	POSTROUTING
nat	POSTROUTING

Kompyuter ilovalari uchun mo'ljallangan paketlarning o'tish tartibi.

<b>Jadvallar</b>	<b>Zanjirlar</b>
mangle	PREROUTING
nat	PREROUTING
mangle	INPUT
filter	INPUT

Kompyuter ilovalaridan yuborilgan paketlarning o'tish tartibi.

<b>Jadvallar</b>	<b>Zanjirlar</b>
mangle	Output
nat	OUTPUT
filter	OUTPUT
mangle	POSTROUTING
nat	POSTROUTING,



1. Paketlarni filtrlashning qanday asosiy parametrlari mavjud?
2. Netfilter-da qanday jadvallar mavjud?
3. Mangle jadvalida qanday zanjirlarni qo'llash mumkin?
4. Netfilter zanjirlari va jadvallari orqali tranzit paketlar qanday tartibda o'tadi?

### 8.3-§. Iptables dasturi

Iptables dasturi netfilter qoidalarini boshqarish uchun mo'ljallangan. U quyidagi imkoniyatlarni beradi:

- qoidalarni qo'shish, o'zgartirish, o'chirish.
- foydalanuvchi zanjirini hosil qilish, o'chirish.
- paketlar hisoblagichini qarab chiqish, o'rnatish va tozalash (yangilash-almashtirish).

Dasturning qaysi jadval bilan ishlashini aniqlash uchun “-t” parametri qo'llaniladi, undan keyin tegishli jadval ko'rsatiladi: filter, nat, mangle. Agar parametr ko'rsatilmagan bo'lsa, dastur filter jadvali bilan ishlaydi.

#### **Iptables dasturining komandalari.**

- A – qoidani qo'shish.
- D – qoidani o'chirish.
- R – qoidani almashtirish.
- I – qoidani qo'yish.
- L – qoidalar ro'yxatini ko'rsatish.
- F- zanjirlar yoki jadvallarni tozalash.
- Z – hisoblagichni nol holatiga keltirish.
- N – foydalanuvchi zanjirini hosil qilish.
- X – foydalanuvchi zanjirini o'chirish.
- R – sukut bo'yicha siyosatini o'rnatish.

Iptables dasturi bajarishi shart bo'lgan amalni aniqlash uchun, unga komandani ko'rsatish zarur bo'ladi.

<b>Komanda</b>	<b>Tavsifi</b>
-A, -append	Zanjir oxiridan qoidani qo'shadi. Komandaning bajarilishi uchun albatta zanjirni ko'rsatish kerak. Misol uchun: iptables -A INPUT -s 172.20.21.100 -j ACCEPT
-D, – delete	Zanjirdan qoidani o'chirish. Qoidani o'chirishning ikkita usuli mavjud: qoidaning nomeri bo'yicha yoki ajratib olishning barcha mezonlarini ko'rsatish bilan. Misol uchun: iptables -D INPUT 1 iptables -D INPUT --dport 80 -j DROP

-R, –replace	Bir qoidani boshqasi bilan almashtirish. Misol uchun: iptables -R INPUT 1 -s 192.168.0.1 -j DROP
-I, –insert	Zanjirning ko‘rsatilgan joyiga qoidani qo‘yish komandasi. Masalan: iptables -I INPUT 1 --dport 80 -j ACCEPT
-L, -list	Agar zanjir nomi ko‘rsatilmasa, bu komanda joriy jadvalning barcha qoidalarini ko‘rsatadi. Agar zanjir nomi ko‘rsatilgan bo‘lsa – ko‘rsatilgan zanjirdagi barcha qoidalarni ko‘rsatadi. Masalan: iptables -L INPUT
-F, -flush	Agar zanjir nomini ko‘rsatmasak, komanda joriy jadvaldan barcha qoidalarni o‘chiradi. Agar zanjir nomi ko‘rsatilsa, ko‘rsatilgan zanjirdan barcha qoidalar o‘chiriladi. Masalan: iptables -F INPUT
-Z, -zero	Jadvaldagi har bir qoidaga ikkita hisoblagich mos keladi: paketlar soni va ushbu qoida qayta ishlovchi baytlar soni. Komanda joriy jadvaldagi hisoblagichlarni nol holatiga keltiradi, yoki, agar zanjir nomi ko‘rsatilsa, berilgan zanjir hisoblagichini nol holatga keltiradi. Misol uchun: iptables -Z INPUT
-N, –new-chain	Joriy jadvalga berilgan nom bilan zanjirni hosil qilish komandasi. Nom unikal bo‘lishi shart va zahiralangan zanjirlar va amallar nomi bilan mos tushmasligi shart. Misol uchun: iptables -N allowed
-X, -delete-chain	Jadvallardan foydalanuvchining zanjirlarini o‘chirish komandasi. Faqat qoidalar yozilmagan zanjirlarni o‘chirish mumkin. Shuningdek, bu zanjirlar boshqa zanjirga aloqasi bo‘lmasligi kerak. Misol uchun:

	iptables - X allowed
-P. -policy	Zanjirlar uchun standart (sukut bo'yicha) siyosatni o'rnatish komandasi. Siyosat hech qanday zanjir qoidalarini ishlatmaydigan paketlar bilan qanday amallar bajarilishi aniqlaydi. Siyosat qiymati sifatida ACCEPT va DROP qiymatlaridan foydalanish mumkin. Misol uchun: iptables -R INPUT DROP

### **Iptables dasturining parametrlari.**

- v – qo'shimcha ma'lumotlarni ko'rsatadi.
  - x – hisoblagichning aniq qiymatini chiqaradi.
  - n – teskari akslantirishni bajarmaydi.
  - line-numbers – zanjirdagi qoida nomerini ko'rsatadi.
  - c – hisoblagich qiymatini o'rnatadi.
- Iptables dasturiga qo'shimcha parametrlarni ko'rsatish mumkin.

<b>Parametrlar</b>	<b>Tavsifi</b>
-v, -verbose	Bu parametr dasturni qo'shimcha ma'lumotlarni chiqarishini ta'minlaydi. Misol uchun, -L komandasi bilan birgalikda nafaqat qoidalarni, balki hisoblagichlar qiymatlarini ham ko'rsatadi.
-x, -exact	Bu parametr dasturni hisoblagichni aniq qiymatini chiqarishini ta'minlaydi.
-n, -numeric	Bu parametr agar dasturning chiqishida IR-adres uchrasa, IR-adresdan kompyuter nomiga teskari akslantirmaslikni ta'minlaydi.
-line-numbers	Bu parametr dasturni zanjirdagi qoida nomerini chiqarishini ta'minlaydi.
-s, -set-counters	Bu parametr zanjirga qoidani qo'shishda hisoblagich qiymatini o'rnatish imkonini beradi.

### **8.4-§. Paketlarni ajratish mezonlari.**

Qoidalarni qo'shishda bu qoidalar ishlashi uchun paketlarni ajratish mezonlarini ko'rsatish zarur. Qoidalarni shartli uchta guruhga

ajratish mumkin:

Umumiy mezonlar – protokol turiga bog‘liq emas va maxsus modullarni yuklashni talab qilmaydi.

Nooshkor mezonlar – protokol turiga bog‘liq va maxsus modullarni yuklashni talab qilmaydi.

Oshkor mezonlar – qo‘llashdan oldin maxsus modullarni oshkor ravishda yuklashni talab qiladi.

Bitta qoidada bir nechta paketlarni ajratish mezonlarini ko‘rsatish mumkin.

Agar qoidada paketlarni ajratib olish mezonlari aniqlanmagan bo‘lsa, demak, mezonlarning ixtiyoriy qiymati o‘rinli bo‘lishini bildiradi.

### **Umumiy mezonlar.**

-r – protokolni aniqlaydi.

-s – manbalarning IP –sini aniqlaydi.

-d – tayinlangan IP-larni aniqlaydi.



-i – kirish interfeysini aniqlaydi.

-o – chiqish interfeysini aniqlaydi.

-f – qismlarga ajratilgan paketning fragmentini aniqlaydi.

Umumiy mezonlar protokol turiga bog‘liq emas va ular uchun maxsus modullarni –m parametri yordamida yuklash talab etilmaydi.

<b>Mezonlar</b>	<b>Tavsifi</b>
-r	Bu mezon protokol turini aniqlashga xizmat qiladi. Qo‘shimcha parametr sifatida nom, protokol nomeri (/etc/protocols fayliga qarang) yoki ALL kalit so‘zini ko‘rsatish kerak. Protokolni ko‘rsatishda mezon qiymatini hisobga oluvchi “!” belgisidan foydalanish mumkin. Misol uchun: -p tcp -p ! icmp
-s	Bu mezon manbaning IP-sini aniqlaydi. Qo‘shimcha parametr sifatida kompyuter yoki tarmoqning IP-adresini ko‘rsatish mumkin. Oxirgi holatda qismtarmoqning niqobini ko‘rsatish zarur bo‘ladi. “!” belgisi parametrning qiymatini hisobga oladi. Masalan: -s 192.168.0.1.

	<p>-s 172.20.21.0/24 -s ! 172.20.21.0/255.255.255.0</p>
-d	<p>Bu mezon yordamida qabul qilib oluvchining IP-adresi aniqlanadi. Qo‘shimcha parametr sifatida kompyuter yoki tarmoqning IP-adresini ko‘rsatish mumkin. Oxirgi holatda qismtarmoqning niqobini ko‘rsatish zarur. “!” belgisi parametrning qiymatini hisobga oladi. Masalan:</p> <p>-d 192.168.0.1 -d 172.20.21.0/24 -d ! 172.20.21.0/255.255.255.0</p>
-i	<p>Paket qabul qilib olingan interfeysni aniqlovchi mezon.</p> <p> <i>Ushbu mezonni faqat INPUT, FORWARD va PREROUTING zanjirlarida qo‘llashga ruxsat beriladi, boshqa ixtiyoriy holatlarda xatolik haqida xabar chaqiriladi</i></p> <p>Agar interfeys nomi + belgisi bilan tugallansa, u holda mezon berilgan satr bilan boshlanuvchi barcha interfeyslarni taqdim etadi. “!” belgisi parametrning qiymatini hisobga oladi. Masalan:</p> <p>-i eth0 -i ppp+ -i ! eth1</p>
-o	<p>Chiquvchi interfeysni aniqlovchi mezon.</p> <p> <i>Ushbu mezonni faqat INPUT, FORWARD va PREROUTING zanjirlarida qo‘llashga ruxsat beriladi, aks hollarda xatolik haqida xabar chaqiriladi.</i></p> <p>Agar interfeys nomi + belgisi bilan tugallansa, u holda mezon berilgan satr bilan boshlanuvchi barcha interfeyslarni taqdim etadi. “!” belgisi parametrning qiymatini hisobga oladi. Masalan:</p> <p>-o eth0</p>



	-o ppp+ -o ! eth1
-f	Mezon qismlarga ajratilgan paketlarning birinchisidan tashqari barchasini aniqlaydi. “!” belgisi parametrning qiymatini hisobga oladi. Masalan: -f ! -f

### Nooshkor mezonlar.

TCP mezonlari:

--sport – manba porti.

--dport – tayinlangan port (qabul qiluvchi).

--tcp-flags - TCP bayroqchalarini aniqlash

--syn – ulanishga so‘rov.

UDP mezonlari:

--sport – manba porti.

--dport – tayinlangan port (qabul qiluvchi).

ICPM mezonlari:

--icmp-type - ICMP paketning turini aniqlaydi.

Oshkor mezonlarni qo‘llash uchun “-r” mezoni yordamida protokolni ko‘rsatish lozim va faqat shundan keyin oshkor mezon aniqlanadi.

TCP mezonlari.

TCP mezonlarini faqat TCP: -r tcp protokolini aniqlagandan keyin qo‘llash mumkin.

Mezonlar	Tavsifi
--sport	Bu mezon paket yuborilgan portni aniqlaydi. Qo‘shimcha parametr sifatida port nomerini yoki xizmatlar nomini ko‘rsatish mumkin (/etc/services faylini qarang). Shuningdek, portlar diapozonini “:” bilan ajratgan holda ko‘rsatish mumkin. “!” belgisi parametrning qiymatini hisobga oladi. Masalan: --sport 1024:65535 --sport 80 --sport ! www
--dport	Mezon tayinlangan portni aniqlaydi. Qo‘shimcha

	<p>parametr sifatida port nomerini yoki xizmatlar nomini ko‘rsatish mumkin (/etc/services faylini qarang). Shuningdek, portlar diapozonini “:” bilan ajratgan holda ko‘rsatish mumkin.</p> <p>“!” belgisi parametrning qiymatini hisobga oladi.</p> <p>Masalan:</p> <pre>--dport 1024:65535 --dport 80 --dport ! www</pre>
--tcp-flags	<p>Mezon TCP paketlarning bayroqchalarini nazorat qilish imkonini beradi. Qo‘shimcha parametr sifatida qiziqtirgan bayroqchani va probel orqali qiymati 1 ga teng bo‘lgan bayroqchalar ko‘rsatiladi. Shuningdek, ALL – barchasi, NONE – hech biri kabi kalit so‘zlaridan foydalanish mumkin.</p> <p>“!” belgisi parametrning qiymatini hisobga oladi.</p> <p>Masalan:</p> <pre>--tcp-flags SYN,ACK,FIN SYN</pre>
--syn	<p>Mezon SYN bayroqchasi o‘rnatilgan va ACK va FIN bayroqchalari tozalangan ulanishda so‘rovlarning mos kelgan TCP paketni aniqlaydi.</p> <p>Misollar:</p> <pre>--syn ! --syn</pre>

**UDP mezonlari.** UDP mezonlarini faqat UDP: -r udp protokoli aniqlangandan keyin qo‘llash mumkin.

Mezonlar	Tavsifi
--sport	<p>Bu mezon paket yuborilgan portni aniqlaydi. Qo‘shimcha parametr sifatida port nomerini yoki xizmatlar nomini ko‘rsatish mumkin (/etc/services faylini qarang). Shuningdek, portlar diapozonini “:” bilan ajratgan holda ko‘rsatish mumkin.</p> <p>“!” belgisi parametrning qiymatini hisobga oladi.</p> <p>Masalan:</p> <pre>--sport 1024:65535 --sport 21 --sport ! www</pre>

--dport	<p>Mezon tayinlangan portni aniqlaydi. Qo‘shimcha parametr sifatida port nomerini yoki xizmatlar nomini ko‘rsatish mumkin (/etc/services faylini qarang). Shuningdek, portlar diapozonini “:” bilan ajratgan holda ko‘rsatish mumkin. “!” belgisi parametrning qiymatini hisobga oladi. Masalan:</p> <pre>--dport 1024:65535 --dport 21</pre>
---------	---

**ICMP mezonlari.** ICMP mezonlarini faqat ICMP: -r icmp protokoli aniqlangandan keyin qo‘llash mumkin.

Hammasi bo‘lib ICMP paketining turini aniqlovchi bitta icmp mezoni mavjud: -icmp-type. Qo‘shimcha parametr sifatida ICMP paketining nomi yoki nomerini ko‘rsatish mumkin.

**Masalan:**

```
--icmp-type echo-request
```

Barcha ICMP paketlarning ro‘yxatini ko‘rish uchun iptables dasturini qo‘llash mumkin:

```
iptables -r icmp --help
```

### 8.5-§. Oshkor mezonlar

**limit** – qoidalarni qayta ishlashlar sonini cheklaydi.

**mac** – paketlarni ajratib olishda mezon sifatida MAC-adresdan foydalanish imkonini beradi.

**multiport** – parametr sifatida bir nechta portlarni qo‘llash imkonini beradi.

**state** – paketlarning holatini aniqlaydi.

Ixtiyoriy oshkor mezonlardan foydalanish uchun “m” parametri yordamida zarur modullarni yuklash kerak bo‘ladi va faqat keyin oshkor mezonni ko‘rsatish mumkin.

**Limit mezoni.**

**--limit-burst** – paketlar sonini aniqlaydi.

**--limit** – vaqt birligini aniqlaydi.

Limit mezonlari ishlov berilgan qoidalar sonini cheklaydi.

Bu mezonni qo‘llashdan oldin “-m limit” parametridan foydalanish zarur.

Mezonlar	Tavsifi
----------	---------

--limit-burst	Bu mezon qoida ishlaydigan paketlar sonini aniqlaydi. Agar bu parametr ko'rsatilmagan bo'lsa, uning qiymati 5 ga teng bo'ladi. Masalan: --limit-burst 1
--limit	Bu mezon vaqt birligi ichida qoidalarga necha marta ishlov berilishini aniqlaydi. Qo'shimcha parametr sifatida vaqt birligini ko'rsatishda quyidagicha foydalanish mumkin: N/s (sekund), N/m (minut), N/h (soat), N/d(kun). N harfining o'rniga albatta sonni kiritish kerak. Misol uchun: --limit 1/s

Misol uchun, paketlarni ajratib olishda quyidagi mezonlar o'rnatilgan:

`-p tcp --dport 80 -m limit --limit 1/s --limit-burst 4 -j ACCEPT`

Bu yozuv TCP (-p tcp) protokolining to'rtta paketi (--limit-burst 4) 80 portiga yo'naltirilgan (--dport 80) sekundiga bir marta (--limit 1/s) qabul qilinishini (-j ACCEPT) bildiradi.

Limit mezonni syn flood va shunga o'xshash hujumlardan himoya qilish uchun qabul qilinadi. Shuningdek, paketlarni Syslog tizimiga uzatishda qabul qilish tavsiya etiladi.

#### **MAC mezon.**

**--mac-source** – paketlar uzatilgan tarmoq tugunining MAC-adresini aniqlaydi.

MAC mezonida paketlarni ajratib olishda ajratib olish mezonni sifatida Ethernet paketining MAC-adresi qo'llaniladi.

<b>Mezon</b>	<b>Tavsifi</b>
--mac-source	Bu mezon paketlar uzatilgan tarmoq tugunining MAC-adresini aniqlaydi. MAC-adreslar XX:XX:XX:XX:XX:XX shaklida ko'rsatilgan bo'lishi shart. Bu mezon faqat PREROUTING, FORWARD va INPUT zanjirlarida ma'noga ega. "!" belgisi parametrning qiymatini tasdiqlaydi. Misol uchun: --mac-source 00:00:00:00:00:01

#### **Multiport mezon.**

**--source-port** – kiruvchi portlar.

**--destination-port** – tayinlangan portlar.

Multiport mezonni parametr sifatida bir nechta portlarni ixtiyoriy ketma-ketlikda ko'rsatish imkonini beradi.

<b>Mezon</b>	<b>Tavsifi</b>
--source-port	Bu mezon chiquvchi portlar ro'yxatini ko'rsatish uchun xizmat qiladi. Mezon yordamida 15 tagacha turli portlarni ko'rsatish mumkin. Portlar nomlari ro'yxatda vergul bilan ajratib ko'rsatilishi kerak, probel qo'yish mumkin emas. Bunday kengaytmanni faqat -r tcp yoki -r udp mezonlari bilan birga qo'llash mumkin. Misol uchun: --source-port 21,23,53
--destination-port	Bu mezon kiruvchi portlar ro'yxatini ko'rsatish uchun xizmat qiladi. Aks hola ushbu mezon --source-port mezoniga o'xshash.

### **State mezonni.**

**--state** – paketning holatini aniqlaydi.

Netfilter ulanishlarni kuzatish imkonini beradi, hatto ICMP va LDP protokollari uchun ham o'rinli. Netfilter ulanish doirasida to'rtta tayanch holatlardan biriga ega bo'lishi mumkin: NEW, ESTABLISHED, RELATED va INVALID. Paketlarning o'tishini boshqarish uchun, state mezonini qo'llagan holda ularning holati asos qilib olinadi.

Ulanishlarni kuzatish yadro maydonida maxsus kod – kuzatuvchi (trassirovka qiluvchi – conntrack) ishlab chiqiladi. Juda ko'p hollarda ulanish haqida odatiy conntrack taqdim etganidan ko'proq batafsil ma'lumotlar talab qilinadi. Shuning uchun ham conntrack o'ziga turli protokollarning qayta ishlovchilarini biriktirib oladi, masalan: TCP, UDP yoki ICMP. Ular yig'ib olgan ma'lumotlar keyinchalik identifikatsiya va ulanishning joriy holatini aniqlash uchun qo'llaniladi. Misol uchun - UDP protokoli bo'yicha ulanish IP-adreslar va manba hamda qabul qiluvchi portlari bo'yicha bir qiymatli aniqlanadi (identifikatsiyalanadi).

Yadroning oldingi versiyalarida paketlarni defragmentatsiyalashni qo'shish/o'chirish imkoniyatlari mavjud edi. Biroq, ulanishni kuzatish netfilter tarkibigi qo'shilgandan keyin bunga ehtiyoj qolmadi. Sababi shundaki, kuzatuvchi (conntrack)

defragmentatsiyani qo'llamasdan unga biriktirilgan funksiyalarni bajara olmaydi va shuning uchun ham u doimo yoniq bo'ladi. Ulanish kuzatuvini o'chirmasdan turib, uni o'chirish taqiqlanadi. Agar kuzatuvchi yoniq bo'lsa, defragmentatsiya doimo bajariladi.

Ulanish PREROUTING zanjirida kuzatuv amalga oshiriladi, faqat paketlar brandmauerdagi lokal jarayonlarda hosil qilingan holatlar bundan mustasno. Bu holatda OUTPUT zanjirida kuzatuv amalga oshiriladi. Bu netfilter barcha hisoblashlarni ushbu zanjir doirasidagi aniq holatlarda amalga oshirishini anglatadi. Brandmauerdagi lokal jarayonlar potokdagi birinchi paketni uzatsa, OUTPUT zanjirida unga NEW holati ta'minlanadi. Javob paketi qaytarilsa, PREROUTING zanjiridagi ulanish holati ESTABLISHED ga o'zgaradi va hakoza. Agar ulanish tashqaridan o'rnatilsa, u holda NEW holati PREROUTING zanjiridagi birinchi paketga ta'minlanadi. Xuddi shunday, nat jadvalidagi PREROUTING va OUTPUT zanjirlari doirasida paketlar holati ham aniqlanadi.

<b>Mezon</b>	<b>Tavsifi</b>
--limit	Bu mezon NEW, ESTABLISHED, RELATED va INVALID holatlarini aniqlaydi. Misol uchun: --state ESTABLISHED,RELATED

Holatlar.

<b>Holatlar</b>	<b>Tavsifi</b>
NEW	NEW belgisi ushbu ulanishdagi birinchi paket hisoblanishi haqidagi xabar. Bu kuzatuvchi moduli ushbu ulanishda ko'rgan birinchi paket ekanligini bildiradi. Misol uchun, agar ushbu ulanish uchun birinchi paket hisoblanuvchi SYN paketi olingan bo'lsa, u NEW halotini oladi. Biroq paket SYN bo'lishi yoki bo'lmasligidan qat'iy nazar NEW holatini oladi.
ESTABLISHED	ESTABLISHED belgisi paket alaqachon o'rnatilgan ulanishga tegishli ekanligini bildiradi.
RELATED	Ulanish ESTABLISHED belgisiga ega bo'lgan boshqa ulanish bilan bog'liq bo'lsa RELATED holatini oladi. Bu shuni bildiradiki, allaqachon ulanish o'rnatilgan ESTABLISHED belgisiga ega bog'lanishda bo'lsa, RELATED holatini oladi. RELATED holatidagi ulanishga misol

	sifatida FTP-data ulanishni olish mumkin.
INVALID	INVALID belgisi paketni tanib bo‘lmasligini va shuning uchun ham unga aniq holatni tayinlash imkoni yo‘qligini anglatadi. Bu turli xil sabablarga ko‘ra kelib chiqishi mumkin, masalan, xotira yetishmasligi yoki ma’lum bo‘lgan ulanishda qandaydir mos kelmaydigan ICMP-xatolik haqidagi xabarni olishda kelib chiqadi.

State mezonni ishlash uchun zarur bo‘lgan filtrlash qoidalari sonini qisqartirish imkonini beradi.

Odatda eng birinchi bo‘lib INPUT va FORWARD zanjirlarida INVALID holatidagi paketlarni qabul qilishni taqiqlovchi qoidalar aniqlanadi:

```
iptables -A INPUT -m state --state INVALID -j DROP
```

```
iptables -A FORWARD -m state --state INVALID -j DROP
```

Keyin allaqachon o‘rnatilgan ulanishdagi paketlarni qabul qilish qoidalarini aniqlanadi:

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Endi kerakli port yoki kompyuterlardan Internet-ga chiqish ruxsati yetarli bo‘ldi, misol uchun:

```
iptables -A FORWARD -r tcp --dport 80 -j ACCEPT
```

### 8.6-§. Amallar va o‘tishlar.

Amallar va o‘tishlar paketga berilgan mezonga mos kelsa, bajarilishi zarur bo‘lgan qoidalarni belgilaydi.

O‘tishlar foydalanuvchi zanjiriga o‘tish imkonini beradi. Bunday holatda “-j” parametridan keyin zanjir nomini ko‘rsatish yetarli bo‘ladi.

Amallar – bu, agar paket berilgan mezonga mos bo‘lsa, bajarilishi zarur bo‘lgan amallarni tavsiflovchi oldindan aniqlangan komanda. Misol uchun, DROP yoki ACCEPT amalini qabul qilish mumkin. Boshqa turdagi amallar ham mavjud, ular haqida keyinroq ko‘rib chiqamiz.

**ACCEPT (paketni qabul qilish) amali.** Agar paket ustida ACCEPT amali bajarilsa, u holda paket zanjir bo‘ylab (va ushbu zanjirga biriktirilgan boshqa chaqiruvchi zanjirlarda) harakatini to‘xtatadi va QABUL QILINGAN hisoblanadi.

**DROP (paketni tashlab yuborish) amali.** Bu amal paketni oddiygina “tashlab yuboradi” va netfilter bu paketning mavjudligi haqida “unutadi”. “Tashlab yuborilgan” paket o‘zining harakatni butunlay to‘xtatadi, ya’ni paket ACCEPT amali bilan bajarilgan holatdagidek boshqa jadvallarga uzatilmaydi.

Shuni esda tutish kerakki, ushbu amal salbiy oqibatlariga olib kelishi mumkin, chunki u server tomonida ham, mijoz tomonida ham ochiq qoldirilgan “o‘lik” tuynuklarni (soketlarni) qoldirishi mumkin. Himoyaning eng yaxshi usuli REJECT amalini qo‘llash hisoblanadi, asosan portlarni skanerlashdan himoyalashda.

**REJECT (paketni tashlab yuborish va xatolik haqida xabar berish) amali.** REJECT amali DROP-dagi kabi holatlarda qo‘llaniladi, lekin DROP-dan farqli ravishda REJECT komandasi paketni uzatuvchi xostga xatolik haqidagi xabarni beradi. REJECT amali faqat INPUT, FORWARD va OUTPUT zanjirlarida (va ularga biriktirilgan zanjirlarda) qo‘llanilishi mumkin.

Parametrlar	Tavsifi
--reject-with	<p>Agar paket berilgan mezon bilan mos kelsa, javobga qanday xabarni berish zarurligini ko‘rsatadi. Paketga REJECT amalini tatbiq qilishda dastlab yuboruvchi xostiga ko‘rsatilgan javob uzatiladi, keyin paket “tashlab yuboriladi”. Quyidagi turdagi javoblarga ruxsat beriladi: icmp-net-unreachable, icmp-host-unreachable, icmp-port-unreachable, icmp-proto-unreachable, icmp-net-prohibited va icmp-host-prohibited. Sukut bo‘yicha port-unreachable xabari beriladi. Faqat TCP protokoli uchun qo‘llaniladigan yana bitta javob turi – tcp-reset. Agar tcp-reset qiymati ko‘rsatilsa, u holda REJECT amali javobga TCP RST paketini beradi. TCP RST paketlari yopiq TCP ulanishlar uchun qo‘llaniladi. Masalan:</p> <p>-j REJECT --reject-with icmp-host-unreachable</p>



## RETURN (foydalanuvchi zanjiridan qaytarish) amali.

RETURN amali qoidalarining joriy zanjiri bo‘ylab paketlarning harakatini to‘xtatadi va chaqiriluvchi zanjirga qaytrishni amalga oshiradi, agar joriy zanjir birlashtirilgan bo‘lsa, yoki, joriy zanjir eng yuqori darajada turgan bo‘lsa (masalan, INPUT), u holda paketga sukut bo‘yicha siyosat tatbiq qilinadi.

**LOG (syslog dasturiga ma’lumot berish) amali.** LOG amali syslog dasturiga ma’lumot beradi.

Ma’lumot sifatida asosan paketning sarlavhasi beriladi. Paketning tarkibi (mazmuni) log faylga tushmaydi. Agar paketning tarkibini berish zarur bo‘lsa, ULOG amalidan foydalanish kerak bo‘ladi. ULOG amalining tavsifi ushbu qo‘llanma tarkibiga kiritilmagan.



*LOG amalini qo‘llashda log tizimiga tushuvchi ma’lumotlarni cheklash uchun limit mezonini yoqish tavsiya etiladi.*

Parametrlar	Tavsifi
--log-level	Berilayotgan xabarning muhimlik darajasini belgilash uchun qo‘llaniladi. Barcha xabarlar kernel vositasi orqali uzatiladi. Misol uchun: -j LOG --log-level debug
--log-prefix	Parametr iptables-ning barcha xabarlari qabul qiladigan matnni beradi. Maxsus prefiksli xabarlarni misol sifatida grep yordamida aniqlash mumkin. Prefiks probellarni qo‘shib hisoblaganda 29 tagacha belgidan iborat bo‘lishi mumkin. Masalan: -j LOG --log-prefix “Alert: ”



1. Qanday komanda yordamida qoidalarni zanjir oxiridan qo‘shish mumkin?
2. Qanday parametr yordamida iptables dasturi ishlaydigan jadvallarni ko‘rsatish mumkin?
3. Paketlarni ajratish mezonlarini qanday turlarga bo‘lish mumkin? Ular nimasi bilan bir-biridan farq qiladi?
4. Limit mezonni nima uchun mo‘ljallangan?
5. Qanday amal yordamida ICMP xabarlarini qaytarish bilan tashlab yuboriladi?

6. Log fayliga ma'lumotlarni yuborish uchun qanday amallar qo'llaniladi?

## 8.7-§. NAT akslantirishlar

NAT akslantirishlar paket sarlavhasidagi uzatuvchi (manba) yoki qabul qiluvchi (tayinlangan) IP-adresini, shuningdek, uzatuvchi yoki qabul qiluvchining portini o'zgartirish imkonini beradi.

Netfilter-da NAT-akslantirishlarni qo'llash mumkin bo'lgan asosiy ikkita asosiy amal mavjud:

- SNAT – uzatuvchi (manba)ning adresi va/yoki portini o'zgartirish.

- DNAT – qabul qiluvchining adresi va/yoki portini o'zgartirish.

SNAT va DNAT lardan tashqari, ularning xususiy hollari ham mavjud:

- MASQUERADE – SNAT-ning xususiy holi.

- REDIRECT – DNAT-ning xususiy holi.

**SHAT.** --to-source – mos maydonini almashtirish kerak bo'lgan IP-adres va/yoki portni aniqlaydi.

Faraz qilaylik, 172.20.21.0/24 tarmog'i mavjud va ushbu tarmoqda joylashgan kompyuterlarni Internet-ga kirishi ruxsatini olish mumkin bo'lsin.

Faqat paketlarni marshrutizatorlardan o'tishiga ruxsat yetarli emas. Klient kompyuteridan uzatiladigan barcha paketlardagi uzatuvchining IP maydonida ichki tarmoqning IP-adresi saqlanadi. Sizning provayderingizning marshrutizatori birinchi bo'lib bunday paketlarni tashlab yuboradi. Hatto agar, bunday paketlar WEB servergacha yetib boradi deb faraz qilsak, server klientga qanday javob berishni bilmaydi. Axir, Internet-dagi marshrutizatorlarda xususiy tarmoqlarga yo'nalishlar hisobga olinmagan.

Bizning vazifamiz marshrutizatorimizdan paketlarning chiqishida xususiy tarmoqning IP-adresini tashqi marshrutizatorning haqiqiy IP interfeysiga almashtirish va tashqi Internet-dan keladigan paketlarning adreslarini xususiy tarmoq adresiga teskari akslantirish mexanizmini ta'minlashdan iborat.

Agar chiquvchi interfeys statik IP-adresga ega bo'lsa, uzatuvchining IP adresi va/yoki portini o'zgartiradigan SNAT amalini qo'llashimiz mumkin. Faraz qilamiz, marshrutizator Internet-ga eth0 interfeysi orqali 195.158.1.1 - IP-adresi bilan ulangan, barcha paketlar ichki tarmoqdagi 172.20.21.1 - IP-adresli kompyuterdan uzatilsin. U holda uzatuvchining IP-adresini o'zgartirish uchun quyidagi komanda qo'llaniladi:

```
iptables -t nat -A POSTROUTING -r tcp -o eth0 -j SNAT --to-source 195.158.1.1
```

NAT akslantirishiga tegishli barcha amallar faqat nat (-t nat) jadvalida bajariladi. SNAT akslantirishni faqat POSTROUTING (-A POSTROUTING) zanjirida bajarilishi mumkin. SNAT amalini qo'llashda albatta protokolni (-r tcp) ko'rsatish kerak bo'ladi. Almashtirish bo'yicha amallar faqatgina Internet-ga chiqadigan, ya'ni eth0 (-o eth0) interfeysidan chiquvchi paketlar ustida amalga oshiriladi. Jo'natuvchining IP adres tashqi interfeysning IP adresiga almashtiriladi - 195.158.1.1 (--to-source 195.158.1.1).

Paketlar ustida qanday amallar bajarilishi mumkinligini quyidagi jadvalda ko'rsatish mumkin:

<i>SNAT-gacha</i>		<i>SNAT-dan keyin</i>	
<i>IPs</i>	<i>Ports</i>	<i>IPs</i>	<i>Ports</i>
172.20.21.	4000	195.158.1.	4000
1		1	

Endi faraz qilaylik, boshqa kompyuterlarda va tarmoqda Internet serveriga yana bitta ulanish va yana 4000 porti bilan ochilgan bo'lsin. Noqulay vaziyatga duch kelamiz. Marshrutizatoridan Internet-ga chiqadigan barcha paketlar ochilgan marshrutda aniq port bilan ko'rinadigan bo'ladi. Va natijada ikkita ulanish bitta va aynan shu portda ochiladi. Bunday bo'lishi mumkin emas. Shuning uchun ham SNAT bajarilishida nafaqat IP-adresni, balki portni ham almashtirish zarur bo'ladi. Masalan, quyidagi ko'rinishda:

```
iptables -t nat -A POSTROUTING -p tcp -o eth0 -j SNAT --to-source 195.158.1.1:30000-35000
```

Endi ulanish ochilayotganda uzatuvchining porti ham almashtiriladi. Buning uchun ko'rsatilganlarning ichidan keyingi bo'sh turgan port tanlanadi:

<i>SNAT-gacha</i>		<i>SNAT-dan keyin</i>	
<i>IPs</i>	<i>Ports</i>	<i>IPs</i>	<i>Ports</i>
172.20.21.	4000	195.158.1.	30000
1		1	
172.20.21.	4000	195.158.1.	30001
2		1	

Netfilter ulanishlarni kuzatib borganligi uchun, barcha orqaga qaytgan paketlarda qabul qiluvchining IP adreslari ichki tarmoq IP-

adresiga almashadi va paket qabul qiluvchi bo'yicha kelib tushadi.

**MASQUERADE.** MASQUERADE amali SNAT akslantirishning xususiy holi hisoblanadi. U RRR ulanishda yoki DHCP serverdan dinamik IP-adres olinadigan holatlarda tavsiya etiladi.

MASQUERADE – berilgan tarmoq interfeysidan IP-adres olish o'rniga uni to'g'ridan-to'g'ri ko'rsatish degani bo'lib, bu SNAT amalidagi --to-source kaliti yordamida bajarilgandek amalga oshiriladi. MASQUERADE amali yaxshi xususiyatga ega – tarmoq interfeysi to'xtatishida ulanishni “unutadi”. SNAT holatida ushbu vaziyatda “unutilgan” ulanishlar haqidagi ma'lumotlar kuzatuvchi jadvalida qoladi va u bir kungacha saqlanadi, bu esa qimmatli xotirani egallab turadi. “Unituvchanlik” dinamik IP-adresga ega bo'lgan tarmoq interfeysi to'xtilishi bilan bog'liq bo'lib, u keyingi yuklanishida boshqa IP-adresni olish ehtimoli mavjud bo'lsada, bu holda ixtiyoriy ulanishlar unutiladi va qoldiq ma'lumotlarni saqlab qolish aqlsizlik bo'ladi.

Ijobiy xususiyatlarga ega bo'lishiga qaramasdan, MASQUERADE amalining afzalliklarini e'tiborga olmaslik lozim, chunki u tizimga juda katta yuklanish beradi.

Misol uchun, oldingi bo'limda tavsiflangan SNAT o'rniga quyidagi komandani qo'llash mumkin:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Bu vaziyatda protokolni ko'rsatish zarurati yo'qligiga e'tibor bering.

**DNAT.** Faraz qilaylik, DNS (172.20.21.0/24) da WEB server (172.20.21.1) joylashgan va ushbu serverga Internet-dan kirishga ruxsatni tashkil etish zarur bo'lsin. Marshrutizatorning (eth0) tashqi interfeysi 195.158.1.1 IP-adresiga ega. Kompyuterdagi www DNS 195.158.1.1 IP-adresiga mos keladi, ya'ni bizning WEB serverga mo'ljallangan barcha paketlar, ushbu IP-adresga 80 portidan o'tadi. Biz ushbu paketlarni 172.20.21.1 kompyuteriga 80 portidan uzatishni tashkil etishimiz kerak bo'ladi.

Bu masalani yechish uchun o'tuvchi paketlarning IP adresi va/yoki portini o'zgartirishga yordam beradigan DNAT amalini qo'llashimiz mumkin.

```
iptables -t nat -A PREROUTING -r tcp --dport 80 -j DNAT --to-destination 172.20.21.1
```

Endi 80 portiga yo'naltirilgan barcha TCP paketlarining

uzatuvchi IP adresini 172.20.21.1 IP-adresiga almashtiriladi. DNAT amalini faqat PREROUTING (marshrutlash haqidagi qaror qabul qilinguncha) zanjirida aniqlash mumkin, IP adres almashtirilgandan keyin paket standart marshrutlash jadvali asosida zarur marshrutga qayta yo'naltiriladi. Xuddi shunday, netfilter ulanishni amalga oshiradi, xotirada SNAT holatidagi jadvalga o'xshash jadval tashkil etadi. Teskari tomonga keluvchi paketlarda uzatuvchining IP adresi avtomatik almashtiriladi.

DNAT amali yordamida, bitta va aynan shu xizmatlarni ko'rsatuvchi bir nechta serverlar o'rtasida ulanishlarni teng taqsimlashni tashkil etish mumkin. Misol uchun, DNS-da uchta WEB server (172.20.21.1 – 172.20.21.3) mavjud va ular o'rtasida ulanishni taqsimlash kerak bo'lsin.

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to-destination 172.20.21.1-172.20.21.3
```

Endi yangi ulanish ayni vaqt momentida eng kam sondagi ochiq ulanishli serverga yo'naltiriladi. To'g'ri, ushbu metodda muhim kamchilik mavjud – agar serverlardan biri ishlashni to'xtatsa, netfilter buni “ko'rmaydi” va yangi ulanishni ishlamayotgan serverga qayta yo'naltirishga harakat qiladi.



1. Dinamik IP-adreslarni olish vaziyatlarida SNAT-ning qanday ko'rinishlaridan foydalanish tavsiya etiladi?
2. SNAT amalini aniqlashda qanday ko'rinishda IP-adreslar va portlar diapozonini ko'rsatish mumkin?
3. Serverlar o'rtasida ulanishlarni teng taqsimlash qanday ko'rinishda bajariladi?

### **8.8-§. Netfilter-ni boshqarish**

Fil'trlash qoidalarini hosil qilgandan keyin, ushbu qoidalarning kompyuterni ishga tushirishda avtomatik yuklanishini tashkil etishimiz kerak.

Iptables bilan birga ikkita dastur qo'yiladi, bu dasturlar yordamida fil'trlash qoidalarini maxsus faylda saqlash va qoidalarni ushbu fayldan tiklash mumkin.

```
iptables-save [-s] > file
```

Dastur standart chiqish uchun joriy qoidalarni aks ettiradi. Agar “-s” parametri ko'rsatilsa, har bir qoida bilan hisoblagich qiymatlari ko'rsatiladi.

iptables-restore [-c] file

Dastur fil'trlashning joriy qoidasini ko'rsatilgan fayldan o'rnatadi. "-s" parametri hisoblagich qiymatlarini tiklaydi.

ASPLinux-da fil'trlash qoidalari haqidagi ma'lumotlar /etc/sysconfig/iptables faylida saqlanadi. Ma'lumotlarni saqlash uchun standart ishga tushiruvchi skriptdan foydalanish mumkin:

```
service iptables save
```

**Ishga tushiruvchi skriptlar.** Kompyuter ishga tushishida fil'trlash qoidalarini avtomatik yoqish uchun va kompyuter o'chirilganda uning avtomatik o'chishi uchun, ishga tushiruvchi skriptlarni yozish yoki tayyorlaridan foydalanish kerak bo'ladi.

RedHat Linux-da ishga tushiruvchi skript mavjud. Lekin SystemV initsializatsiya tizimida ishga tushiruvchi skriptlarga o'zgartirish kiritish taqiqlangan. Firewall-ni dastlabki ishga tushirish uchun kichik dastur yozishga to'g'ri keladi. SuSE Linux-da bunday skript mavjud, lekin uni o'chirish va yangi skript yozish tavsiya etiladi. Slackware Linux-da skript mavjud emas va uni yozish kerak bo'ladi.

Skriptlarni System V tizimining xususiyatlarini hisobga olgan holda yozish kerak bo'ladi. Ya'ni, u kamida quyidagi ikkita funksiyani qo'llashi kerak:

- start – firewall-ni ishga tushirish.
- stop – firewall-ning ishini to'xtatish.

Keltirilgan funksiyalardan tashqari yana quyidagi ikkita funksiyadan foydalanish tavsiya etiladi:

-reset – firewall-ni qandaydir holatidan dastlabki holatga keltirish. Bu funktsiya firewall-ning qoidalarini masofadagi kompyuterda tahrirlash to'g'ri kelgan holatlarda qo'llaniladi.

- init – firewall-ni ishga tushirish uchun qo'llaniladi.

Skriptni yozishni boshlashdan oldin, firewall qanday vazifalarni bajarishi zarurligini to'liq tassavur qilib olishimiz kerak bo'ladi. Eng yaxshisi paketlar o'tishining grafik rejasini chizib olish hisoblanadi. Faqat shundan keyingina skript yozishga kirishish mumkin.

Slackware Linux uchun ishga tushiruvchi skript yozamiz. Bu distributivda BSD ishga tushirish tizimi qo'llanilishiga qaramasdan, skriptni System V ishga tushirish tizimini hisobga olgan holda yozamiz. Shuningdek, bu skriptda kamgina o'zgartirish kiritish bilan SuSE va RedHat Linux distributivlarida ham qo'llash mumkin bo'ladi. RedHat Linux-da faqat oxirgi ikkita funksiyani qo'llash

mumkin bo‘ladi.

Dastlab interpretatorni ko‘rsatamiz va qulaylik uchun bir nechta o‘zgaruvchilarni aniqlaymiz.

```
#!/bin/bash
IPT=/usr/sbin/iptables
IPTR=/usr/sbin/iptables-restore
IPTS=/usr/sbin/iptables-save
```

Start funksiyasi firewall-ni dastlabki ishga tushishini tashkil etishi shart. Buning uchun iptables-restore dasturi va avtomatik hosil qilinadigan /etc/iptables fayli ishlatiladi.

```
start()
{
echo -n "Starting firewall... "
$IPT -F
$IPT -t nat -F
$IPT -t mangle -F
$IPTR -c /etc/iptables
echo "Done"
}
```

Stop funksiyasi joriy filtrlash qoidalarini /etc/iptables fayliga yozishi shart.

```
stop ()
{
echo -n "Stop firewall... "
$IPTS -s > /etc/iptables
echo "Done"
}
```

Reset funksiyasi qoidalarni dastlabki holatga keltirishi kerak. Bu holatda barcha xizmatlarga kirish (ssh bo‘yicha masofadan kirishlardan tashqari) taqiqlangan bo‘lishi kerak.

Yuqorida ta’kidlanganidek, bu funksiyalar filtrlash qoidalarini masofadan turib o‘zgartirish vaqtida qo‘llanishi zarur bo‘ladi. Skriptni reset (quyida tavsiflanadi) parametri bilan ishga tushirish CRON yordamida 15 minutda bir marta amalga oshirilishi kerak. Bu xatolik yuz berganda, filtrlash qoidalari dastlabki holatga kelishi va masofadagi kompyuterdan kirish huquqini olish uchun zarur. Qoidalarni o‘zgartirgandan keyin, skriptni chaqirishni CRON-dan olib



tashlashni unutmashlik kerak.

```
reset()
{
$IPT -F
$IPT -t nat -F
$IPT -t mangle -F
$IPT -P INPUT DROP
$IPT -P FORWARD DROP
#=====
#STATE RULES =====
$IPT -A INPUT -m state --state INVALID -j
DROP
$IPT -A FORWARD -m state --state INVALID -j
DROP
$IPT -A INPUT -m state --state
ESTABLISHED,RELATED -j ACCEPT
$IPT -A FORWARD -m state --state
ESTABLISHED,RELATED -j ACCEPT
# LOCALHOST =====
$IPT -A FORWARD -i ! lo -s 127.0.0.1 -j
DROP
$IPT -A INPUT -i ! lo -s 127.0.0.1 -j DROP
$IPT -A FORWARD -i ! lo -d 127.0.0.1 -j
DROP
$IPT -A INPUT -i ! lo -d 127.0.0.1 -j DROP
$IPT -A INPUT -d 127.0.0.1 -j ACCEPT
$IPT -A INPUT -s 127.0.0.1 -j ACCEPT
# ICMP ENABLED =====
$IPT -A INPUT -p icmp -j ACCEPT
# SSH enable =====
$IPT -A INPUT -p tcp -i eth0 -m state --
state NEW --dport 22 -m
recent --update --seconds 20 -j DROP
$IPT -A INPUT -p tcp -i eth0 -m state --
state NEW --dport 22 -m
recent --set -j ACCEPT
# $IPT -A INPUT -p tcp --dport 22 -j ACCEPT
}
```

Funksiyalarda barcha qoidalar tashlab yuboriladi, standart

siyosat oʻrnatiladi. Keyin state moduli bilan bogʻliq boʻlgan qoidalar oʻrnatiladi. Paketlarning lo interfeysi orqali oʻtishiga ruxsat beriladi. ICMP trafigiga ruxsat beriladi.

Oxirida 22 porti (ssh)ga ulanishga ruxsat beriladi. Lekin ruxsat berish uchun bitta IR-adres bilan ulanishlar sonini cheklashda qoʻllaniladigan recent modulidan foydalaniladi. Keltirilgan misolda 20 sekunda bittadan koʻp boʻlmagan ulanish mavjud. Shunday usul bilan ssh «roboti» boʻyicha parollarni tanlash yopiladi.

Init funksiyasida uning qolgan qoidalarini yozishimiz kerak.

```
init ()
{
echo -n "Init firewall... "
reset
# DNS ENABLE =====
$IPT -A INPUT -p tcp --dport 53 -j ACCEPT
$IPT -A INPUT -p udp --dport 53 -j ACCEPT
# Apache =====
$IPT -A INPUT -p tcp --dport 80 -j ACCEPT
# SMTP =====
$IPT -A INPUT -p tcp --dport 25 -j ACCEPT
# SPOP =====
$IPT -A INPUT -p tcp -i eth0 -m state --
state NEW --dport 995
-m recent --update --seconds 20 -j DROP
$IPT -A INPUT -p tcp -i eth0 -m state --
state NEW --dport 995
-m recent --set -j ACCEPT
# SAVE rules in file ==
$IPTS -c > /etc/iptables
echo "Done"
}
```

Init funksiyasida qanday qoidalardan foydalanish serveringizda yechiladigan masalalarga bogʻliq. Keltirilgan misolda quyidagi serverlarga kirishga ruxsat ochilgan: DNS, WEB, SMTP va SPOP3 (port 995). Funksiya oxirida joriy qoidalarni /etc/iptables fayliga yozish amalga oshiriladi.

Eng oxirida komandalar satri parametrlarida qayta ishlanadigan va mos funksiyani chaqirish kodini yozish kerak boʻladi.

```

case $1 in
start) start ;;
stop) stop ;;
init) init ;;
reset) reset ;;
*)
echo "Usage rc.fw start|stop|init|reset"
exit 88
esac

```

Skript kompyuter ishga tushishida chaqirilish uchun, /etc/rc.d/rc.inet1 faylining tarkibini o'zgartirish kerak.

```

#!/bin/bash
start ()
{
/etc/rc.d/rc.fw start
echo "ifconfig lo 127.0.0.1"
echo "ifconfig lo 127.0.0.1" |
/usr/bin/logger
/sbin/ifconfig lo 127.0.0.1
/sbin/route add -net 127.0.0.0/8 lo
echo "ifconfig eth0 192.168.10.69"
echo "ifconfig eth0 192.168.10.69" |
/usr/bin/logger
/sbin/ifconfig eth0 192.168.10.69
/sbin/route add default gw 192.168.10.1
}
stop ()
{
echo "ifconfig eth0 down"
echo "ifconfig eth0 down" | /usr/bin/logger
/sbin/ifconfig eth0 down
echo "ifconfig lo down"
echo "ifconfig lo down" | /usr/bin/logger
/sbin/ifconfig lo down
/etc/rc.d/rc.fw stop
}
case $1 in
stop) stop ;;
*) start ;;

```

**esac**

Siz tarmoq interfeysini ishga tushirishda o'zingizning parametrlaringizni kiritishingiz kerak. Firewall-ning yuklanishi tarmoq interfeysi ishga tushirilguncha bajarilishiga e'tibor bering. Uning to'xtatilishi esa, interfeysning o'chirilishidan keyin amalga oshadi.

```
/etc/rc.d/rc.fw start
```

```
/etc/rc.d/rc.fw stop
```

## 9-BOB. NFS -TARMOQ FAYL TIZIMI. APACHE VA FTP SERVERLARI

### 9.1-§. NFS – tarmoq fayl tizimi.

Tarmoq fayl tizimi – NFS disk maydonini tarmoq bo‘yicha taqsimlash imkonini beradi. SUN kompaniyasi tomonidan 1985 yilda ishlab chiqilgan.

NFS juda ko‘p komponentlardan tashkil topgan. Linux-da NFS-ni qo‘llash yadro darajasida o‘rnatilgan va shuning uchun ham yadroni NFS-ni qo‘llab quvvatlash bilan yig‘ilganligini ko‘rish mumkin. NFS klient-server prinsipi asosida qurilgan. Kompyuter server yoki klient sifatida, shuningdek, bir vaqtda ikkala vazifani ham bajarishi mumkin.

Linux-da NFS-ning 2 va 3 versiyalari qo‘llaniladi. 2.6.x yadrosida NFS-ning 4 versiyasi sinov tariqasi qo‘shilgan.

NFS – SUN kompaniyasi tomonidan ishlab chiqilgan RPC (Remote Procedure Call) ga asoslanadi. Shuning uchun ham NFS-dan foydalanishdan oldin, RPC-ni qo‘llashni yoqish zarur bo‘ladi.

Ushbu bobning shu bo‘limida NFS klient va serverining ishlashini qanday sozlash haqida gap yuritamiz.

**RPC (Remote Procedure Call).** Port map – dastur so‘rovlarini qabul qiluvchi portlarni aniqlaydi. Rpcinfo - RPC protseduralari bo‘yicha ma’lumotlarni beradi.

RFC serveri RPC klientlari uchun ruxsat berilgan protseduralari to‘plamidan tarkib topgan. Protseduralarni chaqirib olish uchun, klient serverga so‘rov yuboradi, bu so‘rovda protsedura nomi va qo‘shimcha parametrlar ko‘rsatiladi. Server klient tomonidan so‘ralgan protsedurani bajaradi va uning natijasini qaytaradi. Turli apparat platformalarining o‘zaro mosligini ta‘minlash uchun, barcha ma’lumotlar uzatilishidan oldin kompyuterga bog‘liq bo‘lmagan XDR formatiga o‘giriladi. Ma’lumotlarni qabul qilishda esa, lokal tugunlarda foydalanish mumkin bo‘lgan formatga o‘giriladi.

RPC-serverga so‘rov yuborishda RFC protokollarining bir nechta versiyalari mavjud bo‘lganligi sababli, klient nafaqat o‘ziga kerak bo‘lgan protsedura nomerini balki protokol versiyasini ham ko‘rsatishi kerak bo‘ladi.

TCP/IP tarmog‘ida RPC-ni tatbiq qilishda protsedurani tavsiflashda ushbu protsedura so‘rovlarni “qabul qiladigan” port nomeri qo‘shilgan. Transport protokoli sifatida TCP va UDP

protokollaridan foydalanish mumkin. TCP-ga tizim uzatilgan ma'lumotlar hajmi UDP deytagrammalari hajmidan oshib ketgan hollarda murojaat qiladi.

Aniq protseduraga murojaat qilishdan oldin klient protsedura qaysi port orqali so'rovni "eshitishi"ni bilishi kerak. Bu talabning paydo bo'lishiga sabab, protsedurani ishga tushiruvchi dasturlar qaysi port dastur uchun ochiqligini oldindan topa olmaydi. Bu muammoni yechish uchun maxsus portmap dasturi qo'llaniladi. Bu dastur RPC protsedurasini tatbiq qilib, o'zining yuklanishidan keyin RPC-ga zarur bo'ladigan barcha parametrlar va port nomeri haqida xabar beradi. Klientlar protseduradan foydalanishni boshlaganda, ular portmap xizmatiga murojaat qiladi va undan dastur klient so'rovini qabul qiladigan javob beradigan port nomerini oladi.

Slackware Linux-da portmap-ni ishga tushirish uchun etc/rc.d/rc.portmap maxsus skripti qo'llaniladi. Uni bajariluvchi qilish kerak, keyin esa ishga tushirish kerak.

```
chmod a+x /etc/rc.d/rc.portmap  
/etc/rc.d/rc.portmap start
```

Asosiy e'tiborni portmap o'z ishi davomida tcpd dasturini ishlatadi. Bu dastur tcp wrapper deb atalib, begilangan dasturga kirishni cheklash imkonini beradi. Shuning uchun ham, agar /etc/hosts.deny faylda ALL:ALL yozuvi bo'lmasa, /etc/host.allow fayliga quyidagi satrlarni qo'shish kerak bo'ladi:

```
portmap: ALL
```

Kompyuterdan RPC xizmatlari ro'yxatini olish uchun rpcinfo dasturidan foydalanish mumkin:

```
rpcinfo -p kompyuter_IP_adresi
```

Agar faqat portmap ishga tushirilgan bo'lsa, dastur quyidagi satrni chiqaradi:

<b>program</b>	<b>vers</b>	<b>proto</b>	<b>port</b>
100000	2	tcp	111 portmapper
100000	2	udp	111 portmapper

Portmap hamma vaqt 111 portidan so'rovlarni qabul qiladi. Agar firewall orqali RPC-ga kirish ruxsatini olish zarur bo'lsa, RPC protseduralarini tatbiq qiluvchi dasturlar ishga tushirilishida

so'rovlarni qabul qilish uchun ochiq bo'lishi shart bo'lgan port nomerini ko'rsatish mumkin.

## 9.2-§. NFS serveri

Sizning kompyuteringiz NFS serveri vazifasini bajarishi mumkin, bu funksiyani qo'llab-quvvatlashni Linux yadrosida yoqish kerak.

NFS serverining konfiguratsiya fayli - /etc/exports. Bu faylda tarmoqdagi qanday katologlar va kimlar uchun ruxsat berilganligi tavsiflanadi.

/etc/exports faylining formati:

katolog kimga\_mumkin (options) ...

Misol uchun, /etc/exports katologini tarmoq bo'yicha tarqatish zarur bo'lsa, 172.20.21.1 kompyuteridagi klient uchun bu katolog faqat o'qish uchun, 172.20.21.2 kompyuteridagi klient uchun esa to'liq kirish huquqi bilan ochiq bo'lsin. U holda exports fayliga quyidagi satrni qo'shish kerak bo'ladi:

/home/exports 172.20.21.1 (ro) 172.20.21.2 (rw)

Kompyuterlar IP-adreslari o'rniga albatta qismtarmoq niqobini ko'rsatgan holda tarmoq IP-adreslaridan foydalanish mumkin:

172.20.21.0/255.255.255.0(ro)

Agar DNS-da kompyuterlar tavsifi bo'lsa, u holda kompyuter nomlaridan foydalanish ham mumkin. Kompyuter nomlarida "\*" belgisidan foydalanish mumkin.

**Bayroqchalar.** Qavs ichida klientlarni aniqlashda albatta NFS-serveri ishiga ta'sir qiluvchi bayroqchalarni ko'rsatish zarur. Agar bayroqchalar bir nechta bo'lsa, ularni vergul bilan ajratish ko'rsatish kerak bo'ladi.

Demak, NFS – UNIX-ga oid fayl tizimi bo'lib, u fayllarni hosil qilishda barcha muhim atributlarni to'liq beradi, jumladan, ushbu fayl tegishli bo'lgan foydalanuvchining UID-i va guruh GID-ini beradi. Bu tizim xavfsizligiga bog'liq ba'zi muammolarni keltirib chiqarishi mumkin.

Faraz qilamiz, NFS klientiga ulangan katologda UID=1000 bo'lgan user1 foydalanuvchisi fayl yaratgan bo'lsin. Bu fayl NFS-serverida ham UID=1000 foydalanuvchisiga tegishli bo'ladi. Lekin bu foydalanuvchining logini user1 foydalanuvchisidan farq qilishi mumkin. Ya'ni, klientdagi user1 foydalanuvchisiga tegishli bo'lgan fayl, serverda butunlay boshqa foydalanuvchiga tegishli bo'lishi

mumkin. Bunday holatlar yuzaga kelmasligi uchun, serverda va klientdagi bitta UID bitta foydalanuvchiga mos kelishini ta'minlash kerak. Buni amalga oshirish mumkin, masalan, OpenLDAP yordamida.

Root foydalanuvchisi tomonidan yaratilgan fayllar bilan boshqa muammolar paydo bo'ladi. Klient kompyuterida root-ning parolini bilishingiz va ushbu foydalanuvchiga tegishli fayllarni NFS-serveriga ulangan katologda hosil qilishingiz mumkin. Lekin serverning o'zida root foydalanuvchisining parolini bilmaysiz. Bu vaziyat tizim xavfsizligiga jiddiy ta'sir qiluvchi muammo.

Yuqorida sanab o'tilgan muammolarni yechish uchun maxsus bayroqchalardan foydalanish mumkin.

**root\_squash, no\_root.squash va all\_squash bayroqchalari. Nobody foydalanuvchisi.** Root\_squash bayroqchasi sukut bo'yicha o'rnatiladi. U NFS-serverga root foydalanuvchisi fayllari hosil qilinishida, serverdagi ushbu fayllar UID-2 (65534) foydalanuvchisiga tegishli ekanligini ko'rsatadi. Odatda bu foydalanuvchi nobody deb ataladi.



*Linux-da nobody foydalanuvchisi UID 99 ga ega. Shuning uchun ham odatda UID=65534 bo'lgan foydalanuvchi qayd yozuvini nfsnobody hosil qiladi.*

Agar root foydalanuvchisining fayllari hosil qilinganda bu fayllar serverda ushbu foydalanuvchiga tegishli bo'lsa, no\_root\_squash bayrog'i exports faylda aniq belgilanishi kerak.

All\_squash bayrog'i serverning klient kompyuterida ixtiyoriy foydalanuvchi yaratgan fayllarni nobody foydalanuvchisiga berishini ta'minlaydi.

**NFS-serveriga kirishni cheklash.** NFS-serveriga kirishni cheklashning eng yaxshi usuli firewall-dan foydalanish hisoblanadi, shuningdek, tcpd dasturidan foydalanish ham mumkin.

Agar /etc/hosts.deny faylida ALL:ALL satri yoki kirishni cheklovchi boshqa satrlar mavjud bo'lsa, /etc/hosts.allow faylida NFS resurslariga kirishga aniq ruxsat bo'lishi kerak:

```
lockd:ALL
mount:ALL
rquotad:ALL
statd:ALL
```



Ko‘rib turganimizdek, klientlar NFS-serveriga murojaat qilishida ishlatadigan barcha RPC protseduralariga kirishga ruxsat berish kerak.

Agar NFS-serveri ishlayotgan vaqtida /etc/export fayliga o‘zgarishlar kiritilgan bo‘lsa, u holda ushbu o‘zgarishlar kuchga kirishi uchun exportfs dasturini ishga tushirish kerak bo‘ladi:

```
exportfs -ar
```

### **9.3-§. NFS-klienti**

NFS tarmoq diskiga bog‘lanish uchun klientda quyidagilarni amalga oshirish kerak:

- Yadroga NFS-klientning mos versiyasini qo‘llashni yoqish.
- Tarmoq fayl tizimi o‘rnatiladigan ulanish nuqtasini (katologini) yaratish.
- Fayl tizimiga mount dasturi yordamida ulanish.

Mount dasturidan foydalanishda ulanadigan fayl tizimining turini (-t nfs) fayl tizimi joylashgan qurilma sifatida NFS-serveri va ushbu serverga eksport qilinadigan katologning adresi ko‘rsatiladi. Misol uchun, ulanish nuqtasi sifatida /mnt/nfs katologidan foydalaniladi, 172.20.21.2 serverida esa /home/exports katologi eksport qilinadi. Klientda tarmoq fayl tizimiga ulanish uchun, mount dasturi quyidagi parametrlar bilan bajariladi:

```
mount -t nfs 172.20.21.2:/home/exports  
/mnt/nfs
```

Fayl tizimini o‘chirish umount dasturi yordamida amalga oshiriladi:

```
umount /mnt/nfs
```

Agar fayl tizimiga ulanish tizimning ishga tushishida avtomatik bajarilishi zarur bo‘lsa, uning tavsifini /etc/fstab fayliga kiritiladi:

```
172.20.21.2:/home/exports /mnt/nfs nfs rw 0  
0
```

Agar klient-kompyuterida NFS serverining qanday katologlarini eksport qilayotganligini bilish zarur bo‘lsa, eksport qilinayotgan katologlar haqidagi turli xil ma‘lumotlarni olish imkonini beruvchi showmount dasturidan foydalanish kerak.

Misol uchun, 172.20.21.2 NFS serveri qanday katologlarni

eksport qilayotganligini bilish uchun, showmount dasturini quyidagi parametrlar bilan ishga tushirish kerak:

```
showmount -e 172.20.21.2
```



1. Qanday vaziyatlarda portmap dasturining mavjudligi zarur bo'ladi?
2. Qaysi faylda NFS-server eksport qiladigan katalog tavsiflanadi?
3. Agar katalogni eksport qilishda faqat rw parametri aniqlangan bo'lsa, serverda qaysi foydalanuvchi root foydalanuvchisini namoyish etadi?
4. NFS serveridagi `nfs.samtu.uz` saytida joylashgan `/eksport` katalogini kompyuteringizdagi `/mnt/exports` katalogiga ulash uchun ishlatiladigan komandalar satrini yozing.

#### **9.4-§. Apache WEB serveri**

Apache WEB serveri UNIX dunyosida eng ommalashgan serverlardan biri hisoblanadi. Bu mashhurlikka quyidagi sabablarga ko'ra erishilgan:

- Server modulli struktura bo'yicha ishlab chiqilgan. Ya'ni, serverning ishlashidagi barcha xususiyatlar bog'lanuvchi modullar ko'rinishida tatbiq qilingan. Shu sababli yangi imkoniyatlar paydo bo'lishi bilan, ular yangi modullar yordamida osongina qo'shilishi mumkin. Xatoliklar paydo bo'lishi bilan, butun serverni emas, balki xatolik tegishli bo'lgan bitta modulni tuzatish yetarli bo'ladi. Shuningdek, Apache kiruvchi kodlari ochiq API bilan tarqatilganligi sababli, WEB-serverga yangi funksiyalarni qo'shish imkoniyati mavjud.

- Apache-ni turli xil operatsion tizimlarda qo'llash mumkin, jumladan, Windows-da ham. Windows-da 1.3.x versiyasini ishga tushirish imkoni mavjud, lekin unda Windows-ning barcha imkoniyatlari hisobga olinmagan va UNIX-dagi kabi yaxshi ishlaymaydi.

- Apache bilan dinamik WEB-saytlar yaratishda qo'laniladigan turli xil dasturlash tillari integratsiya qilingan. Bular Perl, RNR va Java.

Apache WEB serverining 2-versiyasi jarayonlarni

boshqarishning bir qancha rejimlariga ega. Har bir rejimning o'z moduli mavjud (multiprocessing modules - MRM):

- tarmoqlanuvchi MRM – Apache-ning oldingi versiyasidagi arxitekturani takrorlaydi va so'rovlarga xizmat ko'rsatish uchun ko'plab qism jarayonlarni hosil qiladi.

- potokli MRM – Apache-ning ikkinchi versiyasidan boshlab potoklarni (POSIX threads) qo'llab quvvatlash tatbiq qilingan. Har bir jarayon bir nechta potoklarga ega bo'lishi mumkin. Har bir potok klient so'rovlariga xizmat qiladi. Agar Apache-da 15 ta jarayon ishga tushirilgan bo'lsa va har bir jarayonga 10 potokka egalik qilishga ruxsat berilgan bo'lsa, u holda server  $15 \times 10 = 150$  ta so'rovga bir vaqtda xizmat qilishi mumkin bo'ladi.

- fiksirlangan sondagi qism jarayonli MRM – bu modullar va potoklar orasidagi asosiy farq shundan iboratki, qism jarayonlar soni o'zgarmas va har bir jarayon turli foydalanuvchi va guruhlar identifikatorida ishga tushirilishi mumkin.

- boshqa operatsion tizimlar uchun MRM – boshqa operatsion tizimlarda (Windows, OS/2, Novell) serverning ishlashi uchun bir nechta MRM mavjud.

WEB-serverni yaratishda qanday modul qo'llanilganligini bilish uchun, httpd dasturini modullar ro'yxatini ko'rsatuvchi `-l` parametri bilan ishga tushirish kerak.

Ushbu bo'limda Apache WEB-serverini sozlash bo'yicha umumiy masalalar va shuningdek, virtual xostingni tashkil etish masalalari ko'rib chiqiladi.

**httpd.conf konfiguratsiya fayli.** Apache WEB-server ishga tushirilgandan keyin o'zining konfiguratsiya fayli httpd.conf-ni o'qiydi. Slackware Linux-da bu fayl /etc/apache katologida joylashgan bo'ladi.

## 9.5-§. Apache WEB-serverining parametrlari

Quyida Apache WEB-serverining konfiguratsiya faylida aniqlash mumkin bo'lgan asosiy parametrlari keltirilgan.

Parametrlar	Tavsifi
ServerRoot	Bu parametr WEB-server katologlari daraxtining o'zagini aniqlaydi. Bu katolog html fayllarni saqlash uchun mo'ljallanmagan.

PidFile	WEB-server yuklangandan keyin o'zining PID-ini joylashtiradigan faylga yo'lni aniqlaydi
KeepAlive	Agar parametr On qiymatiga ega bo'lsa – WEB-server bitta ulanishda bir nechta resurslarni berish imkoniyatiga ega bo'ladi.
MaxKeepAliveRequests	Bitta ulanishdagi so'rovlarning maksimal sonini aniqlaydi.
KeepAliveTimeout	Parametr vaqtni sekundlarda aniqlaydi. Agar klient ushbu vaqt mobaynida keyingi resursni so'ramasa server ulanishni ajratadi.
IfModule	Bu direktiva mos modul yuklanganligini tekshiradi va agar u yuklangan bo'lsa, ushbu modulga maxsus parametrlarni qo'llaydi.
StartServers	WEB-serverning yuklanishda hosil bo'lgan qism serverlar soni
MinSpareServers	WEB-serverning minimal nusxalari soni
MaxSpareServers	WEB-serverning maksimal nusxalari soni
MaxClients	Serverda qayta ishlov beriladigan so'rovlar sonini aniqlaydi.
MaxRequestsPerChild	Bitta jarayon qayta ishlashi mumkin bo'lgan so'rovlarning maksimal sonini aniqlaydi.
Listen	WEB-server so'rovlarni qabul qiladigan IP-adres va portni aniqlaydi. Agar IP-adres ko'rsatilmagan bo'lsa – server barcha tarmoq interfeyslaridan so'rovlarni qabul qiladi.
LoadModule	Modul nomi va u joylashgan faylni bog'lash parametri
Include	Tashqi konfiguratsiya fayllari bilan bog'lash direktivasi
Port	WEB-server so'rovlarni qabul qiladigan portni aniqlaydi
User	WEB-server-da ishlash huquqiga ega bo'lgan foydalanuvchilarni aniqlaydi.
Group	WEB-server-da ishlash huquqiga ega bo'lgan guruhni aniqlaydi.
ServerAdmin	Server administratorining Email-ini aniqlaydi

DocumentRoot	html fayllar joylashadigan katalogni aniqlaydi.
--------------	---

**Konteynerlar.** Apache-da ob'ektlar uchun parametrlarni aniqlaydigan turli xil konteynerlar qo'llaniladi:

- Directory – kataloglar.
- DirectoryMatch – oldingi konteynerga o'xshash, lekin parametr sifatida regulyar ifoda qo'llaniladi.
- Files - fayllar.
- FilesMatch – oldingi konteynerga o'xshash, lekin parametr sifatida regulyar ifoda qo'llaniladi.
- Limit – HTTP protokollar metodlari.

**Options.** Directory konteyneri ichida turli opsiyalar aniqlashga yordam beruvchi Options parametri qo'llaniladi. Quyidagi jadvalda parametrlarda qo'llaniladigan opsiyalar keltirilgan:

Opsiyalar	Tavsifi
None	Opsiya yo'q
All	Barcha opsiyalar
ExecCGI	CGI skriptlarining bajarilishiga ruxsat beradi
FollowSymlinks	Server belgili havolalarni bajaradi
Includes	SSI (Server Side Includes) komandalarining bajarilishiga ruxsat beradi
IncludesNoExec	SSI-ni qo'llashda #yexes va #include komandalarini qo'llash taqiqlanadi
Indexes	Agar so'rovda aniq fayl ko'rsatilmagan faqat katalog ko'rsatilgan va ushbu katalogda DirectoryIndex parametri aniqlanmagan yoki sukut bo'yicha fayl topilmasa, server so'ralgan katalogda joylashgan fayl va kataloglar ro'yxatini chiqaradi.
SymLinksIfOwnerMatch	Server belgili havola bo'yicha faqat havolada ko'rsatilgan ob'ekt aynan shu foydalanuvchiga tegishli bo'lsa va uning o'zi ham belgili havola bo'lsagina o'tadi.
MultiViews	Klient tiliga mos hujjatni chiqarish imkonini beradi.

**AllowOverride.** AllowOverride parametri Directory konteyneri

ichida qo'llaniladi. Bu parametr -.htaccess (yoki AccessFileName-da berilgan boshqa ixtiyoriy) faylida qanday parametrlar tavsiflanganligini aniqlaydi.

Quyidagi jadvalda AllowOverride parametrining mumkin bo'lgan qiymatlari ko'rsatilgan:

<b>Parametri</b>	<b>Tavsifi</b>
AuthConfig	Autentifikatsiya va kirishni boshqarish parametrlarini qo'llashga ruxsat berish
FileInfo	Hujjat turini boshqaruvchi parametrlarni qo'llashga ruxsat berish
Indexes	Katologlar indekslarini boshqarish parametrlarini qo'llashga ruxsat
Limit	Kirishni boshqarish parametrlaridan foydalanishga ruxsat berish
Options	Katologlar xususiyatlarini boshqarish parametrlaridan foydalanishga ruxsat berish
None	Ixtiyoriy parametrdan foydalanishni taqiqlash
All	Barcha parametrlardan foydalanishga ruxsat berish

### 9.6-§. Resurslarga kirishni nazorat qilish

Resurslarga kirishni nazorat qilish quyidagi parametrlar yordamida amalga oshiriladi:

- Allow from – kirishga ruxsat beradi.
- Deny from – kirishni taqiqlaydi.
- Order - Allow va Deny parametrlarining tartibini aniqlaydi.

Bu parametrlarni faqat konteynerlarda yoki.htaccess faylida qo'llash mumkin.

**mod\_userdir.c moduli.** Modulni yoqish uchun UserDir parametridan foydalaniladi. Bu parametr uy katologidagi WEB-server foydalanuvchi so'roviga mos html-fayllarni izlaydigan foydalanuvchining katologini aniqlaydi.

Misol uchun, tizimda any foydalanuvchisining uy katologida /home/any katologi mavjud bo'lsin. Unga uy katologida alohida, odatda public\_html katologi hosil qilingan. WEB-serverning konfiguratsiya faylida quyidagi parametr aniqlangan:

UserDir public\_html

U holda ushbu html-fayllarga kirish uchun quyidagi URL-dan foydalanish kerak:

http://host/-any

### Parametrlar (davomi)

Parametri	Tavsifi
DirectoryIndex	Fayl nomini ko'rsatmasdan Serverga murojaat qilganda chiqadigan fayl nomini aniqlaydi. Probellar bilan ajratilgan bir nechta fayllarni ko'rsatish mumkin.
AccessFileName	Foydalanuvchi WEB-serverning parametrlarini oldindan aniqlashi mumkin bo'lgan fayl nomini aniqlaydi.
ErrorLog	Otladka qilish ma'lumotlari joylashadigan faylni aniqlaydi
LogLevel	ErrorLog parametri yordamida aniqlangan faylda joylashgan otladka ma'lumotlari sonini aniqlaydi.
LogFormat	Satr formati va format nomini aniqlaydi.
CustomLog	WEB-serverga kirish haqidagi ma'lumotlar joylashadigan fayl nomini aniqlaydi
Alias	Psevdonimlarni aniqlaydi.
ScriptAlias	Katologlar psevdonimlarini aniqlaydi. Lekin Alias-dan farqli tomoni, server fayl mazmunini ko'rmaydi. U dasturni bajaradi va foydalanuvchiga dastur standart chiqishda ekanligini bildiradi.

**Xatoliklar haqidagi xabar.** Xatoliklar paydo bo'lganda Apache WEB-server xatolik ko'rsatilgan sahifani chiqaradi. Standart sahifalarni o'zingizning shaxsiy sahifangiz bilan almashtirishingiz mumkin. Buning uchun ErrorDocument parametrini qo'llash kerak. Parametr sifatida xatolik nomerini va shuningdek, quyidagilarni ko'rsatish kerak:

- xabar matni
- Html-fayl.
- CGI skripti.
- URL.

Misol uchun:

**ErrorDocument 500 "The server made a boo boo."**

```
ErrorDocument 404 /missing.html
ErrorDocument 404 "/cgi-bin/missing_handler.pl"
ErrorDocument                                     402
http://www.example.com/subs_info .html
```

## 9.7-§. Virtual xosting

Ikki turdagi virtual xosting mavjud:

- IP-adreslarga asoslangan.
- nomlarga asoslangan.

**VirtualHost konteyneri.** Apache-da virtual WEB-saytlarni joylashtirishning keng tarqalgan ikkita metodini qo'llash mumkin:

- IP-adreslarga asoslangan – har bir WEB-sayt alohida IP-adres bilan bog'langan holatda. Bu metod yaxshi variant emas, chunki juda ko'p sondagi IP-adres talab qilinadi.

- nomlarga asoslangan. HTTP 1.1 protokolida klientlarning WEB-serverga so'rov berganida server nomini ko'rsatish ko'zda tutilgan. Ushbu nom bo'yicha Apache WEB serverni bir qiymatli aniqlashi mumkin. Bu metod bitta IP-adres talab qilganligi uchun afzalliklarga ega.

Nomlarga asoslangan virtual xostingdan foydalanishda, NameVirtualHost parametri aniqlangan bo'lishi shart. Har bir virtual xostda shaxsiy VirtualHost konteyneri aniqlangan bo'lishi shart.

Misol uchun, bizning WEB-serverda www.samtuit.uz saytining virtual xostingini ta'minlash zarur bo'lsin.

Birinchi o'rinda, samtuit.uz hududiga mas'ul DNS-serverda www kompyuteri nomini Sizning WEB-serveringiz IP-adresi bilan bog'lovchi A yozuvini aniqlash kerak. WEB-server joylashgan kompyuterda foydalanuvchini qo'shish, masalan, any. Unga uy katalogida (/home/any) yangi katalog public\_html-ni hosil qilish kerak. httpd.conf konfiguratsiya fayliga quyidagi satrlarni qo'shish zarur:

```
NameVirtualHost *:80
<VirtualHost *:80>
    ServerAdmin webmaster@samtuit.uz
    DocumentRoot /home/any/public_html
    ServerName www.samtuit.uz
    ErrorLog /home/any/err/error_log
    CustomLog      /home/any/err/access_log
```



common

`</VirtualHost>`

NameVirtualHost parametri bir marta aniqlanadi. VirtualHost bloklari soni virtual serverlar soni bilan mos bo'lishi kerak. www.samtuit.uz serverning administratori o'zining saytiga ftp bo'yicha kirishga ruxsat oladi. Har bir virtual xost uchun alohida log fayllari bo'lishi tavsiya etiladi. Misolda bu error\_log va access\_log fayllari bilan ko'rsatilgan.

VirtualHost bloki ichida Apache-serverining ixtiyoriy parametrlarini aniqlash mumkin.



*Virtual xosting mexanizmini yoqishdan oldin, bungacha Apache-serveri tomonidan xizmat ko'rsatilgan asosiy saytni VirtualHost konteynerida tavsiflashni unutmag. Agar buni qilmasangiz, saytga kirish mumkin bo'lmaydi.*



1. Apache WEB serverning qanday versiyalarini bilasiz?
2. Apache WEB serverning konfiguratsiya fayli qanday nomlanadi va u qalarda joylashgan bo'ladi?
3. Apache WEB serverning konfiguratsiya faylidagi qanday parametr html fayllarning qalarda joylashganligini aniqlaydi?
4. WEB saytlar nomiga asoslangan virtual xostingni tashqil etishda qaysi parametrni albatta aniqlash kerak bo'ladi?
5. Apache WEB serverning konfiguratsiya faylida qaysi konteyner virtual WEB serverlarni tavsiflash uchun qo'llaniladi?

## 9.8-§. FTP-serveri

Slackware Linux-da ikkita ftp-server qo'yilgan: vsftpd (Very Secure FTP Daemon) va ProFTP. Vsftpd – sozlash juda oson, lekin imkoniyatlarini bo'yicha cheklangan. ProFTP – kuchli server, sozlash yetarlicha oson, FTP serverlarning virtual xostingini yaratishgacha bo'lgan keng imkoniyatlarga ega. Ushbu bo'limda vsftpd-serverini ko'rib chiqamiz.

Serverning konfiguratsiya fayli - /etc/vsftpd.conf.

Quyida ftp-serverning konfiguratsiya faylidan ba'zi parametrlar keltirilgan.

<b>Parametrlar</b>	<b>Tavsifi</b>
anonymous_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, serverga anonim ulanishga ruxsat berilgan bo'ladi.
local_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda tizimning lokal foydalanuvchilari serverga ulanishga ruxsat berilgan bo'ladi.
write_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda yozish amaliga ruxsat berilgan bo'ladi
local_umask	Hosil qilinadigan fayllar uchun niqob qiymatini aniqlash parametri
anon_upload_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda anonim foydalanuvchilarga ftp-serverdan fayllarni ko'chirib olishga ruxsat berilgan bo'ladi.
anon_mkdir_write_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda anonim foydalanuvchilarga ftp-serverdan yangi katalog ochishga ruxsat berilgan bo'ladi.
dirmessage_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda, ftp-serverning kataloglaridagi fayllardan foydalanishga ruxsat berilgan bo'ladi.
xferlog_enable	Agar ushbu parametarning qiymati YES-ga teng bo'lsa, u holda, log faylga ko'chirib olingan va yuklangan fayllar haqidagi ma'lumotlar bilan kirishga ruxsat berilgan bo'ladi.
xferlog_file	Parametr ko'chirib olingan va yuklangan fayllar haqidagi ma'lumotli faylning joylashish o'rnini aniqlaydi. Agar bu parametr aniqlanmagan bo'lsa, sukut bo'yicha qiymati /var/log/vsftpd.log bo'ladi.
ftpd_banner	Serverga ulangan ftp-klientlarga chiqadigan satrni aniqlash parametri.
tcp_wrappers	Agar ushbu parametarning qiymati YES-ga

	teng bo'lsa, u holda, TCP wrapper-ni qo'llashga ruxsat berilgan bo'ladi (/etc/hosts.allow va /etc/hosts.deny konfiguratsiya fayllari).
chroot_local_user	Agar ushbu parametrning qiymati YES-ga teng bo'lsa, u holda, lokal foydalanuvchilar serverga kirgan vaqtda uy katologidan tashqariga chiqa olmaydi. Anonim foydalanuvchilar xech qachon lokal foydalanuvchilar ko'rsatgan uy katologidan tashqari chiqa olmaydi.

FTP-serverning asosiy konfiguratsiya faylidan tashqari qo'shimcha fayllari ham mavjud.

/etc/vsftpd.ftpusers – bu faylning har bir satrida alohida ftp-serverga kirishi taqiqlangan foydalanuvchilar tavsiflanadi. Fayl mavjud bo'lmasligi ham mumkin.

Agar userlist\_deny parametrining qiymati YES-ga teng bo'lsa, u holda /etc/vsftpd.ftpusers faylida keltirilgan foydalanuvchilarning sistemaga kirishi taqiqlanadi. Sababi ulardan parol so'ralmaydi va server login kiritilgandan keyin darhol ulanishni ajratadi. Agar userlist\_deny parametrining qiymati NO-ga teng bo'lsa, u holda ushbu faylda tizimga kirishi mumkin bo'lgan barcha foydalanuvchilarni aniq ko'rsatishimiz kerak bo'ladi.

FTP-server ishiga ta'sir qiluvchi yana bir konfiguratsiya fayli /etc/shells. Ushbu faylning har bir satrida /etc/passwd-faylida foydalanuvchi shell-i sifatida o'rnatilishi mumkin bo'lgan dasturlar tavsiflanadi. Agar foydalanuvchi ftp-serverga ulanishga harakat qiladigan shell bu faylda mavjud bo'lmasa, ftp-server foydalanuvchilar kirishini inkor qiladi. Bu xususiyat WEB-serverlarning virtual xostingini tashkil qilishda juda qulay, chunki foydalanuvchi o'z uy katologiga FTP-serveri orqali kirishni bilishi shart, lekin tizimga ssh yoki shunga o'xshash dasturlar orqali kirish imkoniyatini bilishi shart emas. /etc/shells fayliga /sbin/false dasturini qo'shish yetarli. Aynan shu dasturni foydalanuvchi shell-i sifatida /etc/passwd fayliga yozish kerak. Shundan so'ng foydalanuvchi o'zining fayllariga faqat ftp-serveri orqali kirish imkoniyatlariga ega bo'ladi.

## 9.9-§. FTP-serverini ishga tushirish

Slackware Linux-da serverni ishga tushirish tarmoq superserveri inetd-dan foydalanish bilan amalga oshiriladi.

**FTP-serverga anonim kirish.** Agar har bir foydalanuvchi kirishi mumkin bo'lgan ftp-serverni tashkil qilish zarurati bo'lsa, quyidagi amallarni bajarish kerak bo'ladi:

1. Konfiguratsiya fayli /etc/vsftpd/vsftpd.conf faylidagi anonymous\_enable parametriga YES qiymatini ta'minlash.

2. Agar anonim foydalanuvchilarga ham ftp-serverdan fayllarni ko'chirib olishga ruxsat bermoqchi bo'lsak, anon\_upload\_enable parametriga YES qiymatini ta'minlash kerak.

3. Agar anonim foydalanuvchilarga ftp-serverda katalog tashkil etishga ruxsat berish kerak bo'lsa, u holda anon\_mkdir\_write\_enable parametriga YES qiymatini ta'minlash kerak.

Anonim kiruvchi foydalanuvchilar ftp-ning lokal foydalanuvchilarida ko'rinishiga e'tibor bering. Serverga ulanishdan keyin ular ushbu lokal foydalanuvchilarning uy katalogiga kiradi va undan kam foydalanmaydi.



1. Slackware Linux distributivida qanday ftp-serveri o'rnatilgan?

2. VSFTPD FTP-serverining konfiguratsiya fayllari qanday ataladi va ular qaerda joylashgan bo'ladi?

3. Asosiy konfiguratsiya faylidan boshqa qanday fayllar FTP-serverning ishlashiga ta'sir qiladi?

4. FTP-serverga anonim kirishdan foydalanadigan foydalanuvchilar qanday lokal foydalanuvchilarni tasvirlaydi?

5. Lokal foydalanuvchilar o'zining fayllariga FTP-server orqali kirish imkoniyatiga ega bo'lishi uchun qanday shell-ni aniqlashi kerak?

## 10-BOB. LABORATORIYA MASHG‘ULOTLARI

### 1-laboratoriya ishi. Slackware Linux-da tarmoq sozlash

**Ishning maqsadi.** Slackware **Linux-da** tarmoq interfeysi va martshrutlash jadvalini konfiguratsiyalashni o‘rganish.

**Kiruvchi ma’lumotlar.** Quyida tarmoq interfeysining asosiy parametrlari va jimlik holati bo‘yicha marshrutlarni yozing.

<i>eth0 interfeysi</i>	
IP adres	
Maska	
Efir (keng tarqatish) adresi	
Modul nomi	
<i>eth1 interfeysi</i>	
IP adres	
Maska	
Efir (keng tarqatish) adresi	
Modul nomi	
<b>Jimlik holati bo‘yicha marshrut</b>	
IP adres	

<b>Topshiriqlar</b>	<b>Tavsifi</b>
1. Modullarni yuklashni sozlash	1. /etc/rc.d/rc.netdevice faylini tahrirlash uchun oching va tarmoq kartasi modulini yuklash satrini qo‘shing
2. /etc/rc.d/rc.inet1 faylini o‘zgartirish	1. /etc/rc.d/rc.inet1 faylini tahrirlash uchun oching 2. Interpretator ko‘rsatilgan birinchi satrdan boshqa barcha satrlarni o‘chiring. 3. Faylga 127.0.0.0 tarmog‘i qo‘shilgan barcha tarmoq interfeyslarni va jimlik holati bo‘yicha marshrut qo‘shishni tavsiflovchi beshta satrni qo‘shing.

3. Ishchi holatini tekshirish	<ol style="list-style-type: none"><li>1. barcha ishlovchi tarmoq interfeyslarini qo‘shing.</li><li>2. Barcha tarmoq kartalarini ishga tushiring (rmmmod dasturi)</li><li>3. /etc/rc.d/rc.netdevice skriptini bajaring va modullar ro‘yxatiga qarang. Ro‘yxatda barcha tarmoq kartalarining modullari bo‘lishi shart.</li><li>4. /etc/rc.d/rc.inet1 skriptini bajaring. Marshrutlash jadvaliga qarang. Unda barcha tarmoq interfeyslariga ulangan marshrutlar tavsifi va jimlik holati bo‘yicha marshrut bo‘lishi shart.</li></ol>
-------------------------------	---

## 2-laboratoriya ishi. Root foydalanuvchisining parolini o'zgartirish

### Ishning maqsadi.

Root foydalanuvchisining paroli yoddan chiqqanda uni o'zgartirishni o'rganish.

Topshiriq	Tavsifi
1. Standart initsializatsiya tizimini o'zgartirish bilan yadroni yuklash	1. Kompyuterni qo'shish 2. Yuklovchidan Linux yadrosiga mos keluvchi satrni tanlang va unga quyidagi parametрни qo'shing: <b>init=/bin/bash</b> Yadroning yuklanishi amalga oshadi va bash interpretatori ishga tushadi.
2. root foydalanuvchisining parolini o'zgartirish	Ayni paytda Siz komandalar satrini root foydalanuvchisining huquqi bilan boshqarish va ildiz katologga minimal komandalar to'plami bilan ulanish imkoniga egasiz. 1. Ildiz fayl tizimini rw rejimida qayta o'rnatish: <b>mount -o rw,remount /</b> 2. Superfoydalanuvchining parolini almashtiring: <b>passwd</b> 3. Diskdagi fayl tizimi buferini tozalang: <b>Sync</b> 4. Kompyuterni qayta yuklang: <b>Ctrl+Alt+Del</b>

### 3-laboratoriya ishi. IDE kontrollerlarni sozlash

**Ishning maqsadi:** IDE kontroller parametrlarini qo'lda sozlashni o'rganish.

Topshiriq	Mazmuni
1. Joriy o'rnatilganlarni ko'rish	<p>1. Siz root foydalanuvchi huquqida ishlayotganingizga ishonch hosil qiling.</p> <p>2. /dev/hda interfeysining joriy parametrlarini ko'ring hdparm /dev/had</p> <p>3. /dev/hda interfeysiga bog'langan (tizimning yuklanish vaqtida olingan) qurilma parametrlarini ko'rib chiqing hdparm -i /dev/hda</p> <p>4. /dev/hda interfeysiga bog'langan qurilmalarning joriy parametrlarini qo'rib chiqing hdparm -I /dev/hda   less</p> <p>5. /dev/hda-ga bog'langan qurilmalarning tezkorligini tekshiring: hdparm -tT /dev/hda</p> <p>Oxirgi topshiriqni uch marta qayta tekshiring</p>
2. Interfeys parametrlarini sozlash	<p>1. UDMA va "R/W multiple sector transfer" ning maksimal qiymatlarini ko'rib chiqing: hdparm -I /dev/hda   grep DMA: hdparm -I /dev/hda   grep R/W</p> <p>2. UDMA ga qo'yiladigan maksimal qiymat va sektorlar sonini almashtirish uchun quyidagi satrni bajarib ko'ring: hdparm -d1 -c3 -m &lt;sektorlar soni&gt; -X&lt;64+UDMA rejimi&gt; /dev/hda</p> <p>3. Interfeys konfiguratsiyasini tekshirib ko'ring:</p>



	<p>hdparm /dev/hda</p> <p>4. Qurilmaning tezkorligini testlang: hdparm -tT /dev/hda</p>
<p>3. Yuklovchi (initsializatsiya) tizim interfeysi parametrlarini sozlash</p>	<p>1. /etc/rs.d/rc.S faylini tahrirlash uchun oching.</p> <p>2. Faylning oxiriga quyidagi satrning qo'shing: hdparm -d1 -m sektorlar_soni -c3 \ -X udma_rejimi /dev/hda</p> <p>3. Faylni saqlang va tizimni qayta yuklang.</p> <p>4. Tizimga root foydalanuvchisi sifatida kiring.</p> <p>5. Joriy tizim parametrlarini tekshiring</p>

#### 4- laboratoriya ishi. Disk maydonidan foydalanishni cheklash sozlamalari

**Ishning maqsadi.** Disk maydonidan foydalanishni cheklash sozlamalarini o'rgnaish

Topshiriqlar	Tavsifi
<p>1. Kvotalash mexanizmini qo'shish</p>	<p>1. root foydalanuvchisi huquqida ishlayotganingizga ishonch hosil qiling            2. quota_v2 modulini yuklang:            modproba quota_v2            Litsenziya yo'qligi to'g'risidagi xabarga e'tibor bermang.            3. /etc/fstab faylida /home fayl tizimining tavsifiga ulash parametrlarini qo'shing:            usrquota, grpquota            4. Fayl tizimini o'chiring va qo'shing:            umount /home            mount /home            5. Kvota haqida ma'lumotlar saqlanadigan faylni hosil qiling:            touch /home/{aquota.user, aquota.group}            chmod 600 /home/{aquota.user, aquota.group}            6. user2 foydalanuvchisi uchun hisob yozuvini hosil qiling:            adduser user2            7. quotacheck dasturini ishga tushiring:            quotacheck -aavg            8. Kvotalash mexanizmini qo'shing:            quotaon -avg</p>
<p>2. users foydalanuvchilari uchun kvotani sozlash</p>	<p>1. VISUAL o'zgaruvchisini aniqlang:            export VISUAL=/usr/bin/mcredit            2. Kvotani tahrirlash dasturini ishga tushiring:            equota -u user2            3. blocks bo'limining soft va hard maydoniga 100 ni yozing.</p>

	4. Faylni saqlang va redaktordan chiqing.
3. Ishchi holatni tekshirish	<p>1. Boshqa virtual terminaldan tizimga user2 foydalanuvchi sifatida kiring.</p> <p>2. Faylni 200 blokli o'lchamda hosil qilishga harakat qiling:  <code>dd if=/dev/zero of=file bs=1024 count=200</code></p> <p>Ekranda kvota ajratilganligi haqida ogohlantiruvchi axborot chiqishi kerak.</p> <p>3. Hosil qilingan faylning o'lchamini tekshiring.</p> <p>4. Kvotaning joriy holatini tekshiring:  <code>quota</code></p> <p>5. user2 foydalanuvchi hisob yozuvidan chiqing.</p> <p>6. Tizimga root foydalanuvchisi huquqi bilan kiring.</p> <p>7. user2 foydalanuvchisining kvotasini tekshiring:  <code>quota user2</code></p>

## 5-laboratoriya ishi. Syslog tizimini sozlash.

**Ishning maqsadi:** Syslog tizimidan foydalanishni o‘rganish.

Topshiriq	Tavsifi
1. Yangi log faylini qo‘shish	<p>1. root foydalanuvchisi huquqida ishlayotganingizga ishonch hosil qiling.</p> <p>2. Matn redaktorida <code>/etc/syslog.conf</code> faylini oching.</p> <p>3. Quyidagi satrni toping  <code>authpriv.* -/var/log/secure</code>                      va uni quyidagicha o‘zgartiring  <code>authpriv.*; auth.* -/var/log/secure</code></p> <p>4. Birdaniga ushbu satrdan keyin quyidagi satrni qo‘shing:  <code>auth,authpriv.=notice -/var/log/auth.notice</code>                      Muhimlik darajasi notice bo‘lgan autentifikatsiya tizimining barcha ma’lumotlari nafaqat secure fayliga, balki <code>auth.notice</code> fayliga ham yoziladi.</p> <p>5. Shu vaqtning o‘zida <code>mail.* - /var/log/maillog</code> satridan keyin <code>mail.* - /dev/tty9</code> satrini qo‘shing.                      Endi pochta tizimining barcha fayllari <code>tty9</code> virtual terminaliga nusxalanadi.</p> <p>6. <code>syslogd</code> dasturiga o‘zining konfiguratsiya fayli bilan ishlashi uchun HUP signalini yuboring.  <code>killall -HUP syslogd</code></p> <p>7. <code>/var/log</code> katologida <code>auth.notice</code> fayli hosil bo‘lganligiga ishonch hosil qiling.</p>
2. syslog tizimining yangi konfiguratsiyasining ishchi holatini tekshirish	<p>1. Yangi foydalanuvchi <code>user1</code> ni <code>user1</code> paroli bilan qo‘shing.</p> <p>2. Boshqa virtual konsolda <code>user1</code> foydalanuvchisiga dastlab noto‘g‘ri parol bilan kirinishga urinib ko‘ring, keyin to‘g‘ri parol bilan kiring.</p> <p>3. Ushbu seansning o‘zida <code>user1</code> foydalanuvchisining parolini o‘zgartiring.</p> <p>4. <code>su</code> dasturi yordamida root</p>

	<p>foydalanuvchisi imtiyozlarini oling.</p> <p>5. su dasturidan chiqing.</p> <p>6. user1 foydalanuvchisi seansini tugating.</p> <p>7. Tizimga root foydalanuvchisi sifatida kiring</p> <p>8. /var/log/secure va /var/log/auth.notice fayllarining mazmunini ko‘rib chiqing</p> <pre>tail /var/log/secure tail /var/log/auth.notice</pre>
--	--

## 6-laboratoriya ishi. auth.notice faylini nazorat qilish uchun skript yaratish va qo‘llash

**Ishning maqsadi.** Tizimga root foydalanuvchisi sifatida kirishni nazorat qilish uchun skript yozish.

Topshiriq	Tavsifi
1. Skript yaratish	<p>1. root foydalanuvchisi huquqi bilan ishlayotganligingizga ishonch hosil qiling.</p> <p>2. Quyidagi mazmunda /usr/local/sbin/mailalert faylini hosil qiling:</p> <pre>#!/bin/bash LOG=/var/log/auth.notice if [ -s \$LOG ]; then     if cat \$LOG   grep "ROOT LOGIN" &gt; /dev/null     then         cat \$LOG   mail -s "Alert" root     fi     echo -n &gt; \$LOG fi</pre> <p>3. Bu faylni ishga tushiring</p>
2. Skriptning ishlashini tekshirish	<p>1. mailalert dasturini bajarilishi uchun yuklang. Natijada /var/log/auth.notice faylining mazmunidan iborat xat olishingiz kerak.</p> <p>2. /var/log/auth.notice faylining mazmunini ko‘rib chiqing.</p>

## 7-laboratoriya ishi. Cron-dan foydalanish.

**Ishning maqsadi:** Cron dasturidan foydalanib dasturlarning bajarilishi jadvalini tuzishni o'rganish.

Topshiriq	Tavsifi
1. mailalert skriptini jadval bo'yicha yuklash	1. root foydalanuvchisi huquqi bilan ishlayotganingizga ishonch hosil qiling 2. VISUAL muhiti o'zgaruvchilarini Siz foydalanayotgan redaktorga yo'l ko'rsatilgan holda aniqlang. 3. crontab –ye dasturini ishga tushiring. 4. mailalert dasturini yuklashni tavsiflovchi satrni qo'shing. <i>Qo'shilishi kerak bo'lgan satrni o'qituvchingizdan so'rang.</i>
2. Dasturni ikki minutda bir martalik davr bilan yuklash	1. Faylga quyidagi satrni qo'shing */2 * * * * /bin/data >> /var/log/data /var/log/data fayli log fayllarini o'rnini almashtirish (rotatsiya) sozlamalari bo'yicha laboratoriya ishi uchun keyinchalik kerak bo'ladi. 2. Redaktordan saqlash bilan chiqing. 3. Ikki minutdan keyin /var/log/data fayli hosil bo'lganligiga ishonch hosil qiling.

## 8-laboratoriya ishi. At tizimini qo'llash

**Ishning maqsadi:** Aniq vaqtda topshiriqlarning bajarilishini tashkillashtirish uchun at tizimini qo'llashni o'rganish.

<b>Topshiriq</b>	<b>Tavsifi</b>
1. Topshiriqni bajarilishi uchun navbatga qo'yish	1. at dasturini ishga tushiring: at now +7 minutes 2. «>» belgisidan keyin quyidagi komandani kiriting: date 3. Enter tugmasini bosib va keyingi komandani kiriting: ls -l 4. Enter tugmasini bosib 5. Ctrl+D tugmalar kombinatsiyasini bosib.
2. Bajarilishi uchun navbatning mazmuni ko'rish	1. atq komandasini bajaring 2. /var/spool/atjobs katologining mazmunini va unda joylashgan fayllarni ko'rib chiqing.
3. at tizimining ishchi holatini tekshirish	Atd dasturining ishlashida nechta komanda mavjud, o'z ishingizning natijalarini standart bo'lmagan chiqishlarga chiqaring, atd bajarilish navbatini qo'ygan foydalanuvchilarga xat (xabar) yuboring.



## 9-laboratoriya ishi. Logrotate dasturini qo'llash.

**Ishning maqsadi.** Fayllar rotatsiyasini sozlashni o'rganish.

<b>Topshiriq</b>	<b>Tavsifi</b>
1. Konfiguratsiya faylini hosil qilish	<p>1. root foydalanuvchisi huquqida ishlayotganingizga ishonch hosil qiling.</p> <p>2. /etc/logrotate.d/date faylini hosil qiling. Ushbu faylga /var/log/date faylini rotatsiya qilish jarayonini tavsiflaymiz.</p> <p>3. Faylga quyidagi satrlarni qo'shing:</p> <pre>/var/log/date { mail root mailfirst size=100 }</pre> <p>va uni saqlang.</p>
2. Rotatsiya jarayonini ishga tushirish	<p>Fayllar rotatsiyasini logrotate dasturining jadval bo'yicha ishga tushishini kutib o'tirmasdan qo'lda ishga tushirish mumkin.</p> <p>1. Rotatsiya jarayonini bajarish uchun quyidagi komandalarni bajaring:</p> <pre>logrotate /etc/logrotate.conf</pre> <p>2. /var/log katologining mazmunini ko'rib chiqing.</p>

## 10-laboratoriya ishi. Inetd superserverini qo'llash

**Ishning maqsadi:** Inetd tarmoq superserverini sozlashni o'rganish.

Topshiriq	Tavsifi
1. ftp, ror3 va finger serverlarini qo'shish (ishga tushirish)	<p>1. root foylanuvchisi huquqida ishlayotganingizga ishonch hosil qiling.</p> <p>2. Tahrirlash uchun etc/inetd.conf faylini oching.</p> <p>3. Fayldan quyidagi satrni toping: #ftp stream tcp nowait root /usr/sbin/tcpd vsftpd Satr boshidagi izoh belgisini o'chiring. Shunday usul bilan ftp server ishga tushiriladi.</p> <p>4. Quyidagi satrni toping: #pop3 stream tcp nowait root /usr/sbin/tcpd /usr/sbin/pops3d Satr boshidagi izoh belgisini o'chiring. pop3 serveri ishga tushiriladi</p> <p>5. Quyidagi satrni toping: #finger stream tcp nowait nobody /usr/sbin/tcpd in.fingerd -u Satr boshidagi izoh belgisini o'chiring. finger serveri ishga tushiriladi</p> <p>6. inetd serverini o'zining konfiguratsiya faylini o'qishi uchun ishga tushiring: killall -HUP inetd</p> <p>7. Tekshirish uchun 21, 79 va 110 portlari ochiqligiga ishonch hosil qiling netstat -nlp</p>
2. finger serverining ishchi holatini tekshirish	<p>1. finger serveridan ma'lumot olishga urinib ko'ring: finger root@ip_qo'shni_adres ip_qo'shni_adres o'rniga siz bilan juftlikda ishlayotgan qo'shni IP adresini qo'ying.</p>
3. Servisga kirishni tcpd yordamida cheklash	<p>1. /etc/hosts.deny faylini tahrirlash uchun oching. Unga quyidagi satrni qo'shing: ALL:ALL</p> <p>2. /etc/hosts.allow faylini tahrirlash uchun oching. Unga quyidagi satrni qo'shing:</p>

	in.fingerd : ip_qo'shni_adres
--	-------------------------------

3. Sizning qo'shningiz finger serveridan ma'lumot olishga urinib ko'ring. Keyin boshqa juftlikda ishlovchi serverdan.

## 11-laboratoriya ishi. Tinglovchilar kompyuterida DNS serverni sozlash

**Ishning maqsadi:** DNS serverlarni sozlash va boshqarishni o'rganish.



*Ishni bajarish uchun juftlikda ishlaymiz. Juft nomerli kompyuterlarda DNS serverni sozlaymiz. Toq nomerli kompyuterlarda esa pochta serverini sozlaymiz.*

**Kiruvchi ma'lumotlar.** Siz o'qituvchi tomonidan boshqarish huquqi berilgan domenni qo'llovchi DNS serverni sozlashingiz kerak.

Domen nomini yozing \_\_\_\_\_

Hududni tavsiflash faylida pochta serveri sifatida ishlovchi toq nomerli kompyuterlarni ko'rsatish uchun MX yozuvni aniqlash kerak.

Laboratoriya ishi natijasida Siz faqat o'z kompyuteringizning nomini akslantirishingiz shart.

Akslantirishni tekshirish maqsadida ping dasturidan foydalaning.

Topshiriqlar	Mazmuni
1. named.conf fayliga hududni tavsiflash faylini qo'shish	1. Tahrirlash uchun /etc/named.conf faylini oching va unga hudud parametrlarini aniqlash uchun zone ko'rsatmasini qo'shing. Quyida bu faylga qo'shilgan satrlarni yozing. _____ _____ _____
2. Hududni tavsiflash faylini hosil qilish	1. /var/named katologida hududni tavsiflash faylini hosil qiling. Quyida ushbu faylga joylashtirgan satrlarni yozing. _____ _____ _____
3. Sintaksis xatoliklarni tekshirish	1. named-checkconf dasturi yordamida named.conf faylini sintaksis xatoliklarga tekshiring. 2. named-checkzone dasturi yordamida hududni tavsiflash faylidagi sintaksis

	xatoliklarni tekshiring.
4. DNS serverni yuklash	<ol style="list-style-type: none"> <li>1. DNS serverni ishga tushiring named</li> <li>2. Jarayonlar ro'yxatiga qarang va named dasturi ushbu ro'yxatda bor yo'qligiga ishonch hosil qiling</li> <li>3. named dasturi /var/log/messages log faylining oxiriga qanday ma'lumotni qo'shganini ko'ring</li> </ol>
5. DNS klientni sozlash	1. Ikkala kompyuterda ham sizning DNS serveringizni qo'llay boshlaydigan DNS klientni sozlang.
6. Serverning ishchi holatini tekshirish	1. ping dasturini qo'shni kompyuterning FQDN nomini ko'rsatish orqali bajaring.
7. Yangi konfiguratsiya faylida kompyuter nomini o'zgartirib, serverni qayta yuklang.	<p>Kompyuter nomini o'zgartirgandan keyin, yangi nomni konfiguratsiya fayliga yozish va kompyuter nomining o'zgarishiga ta'sir etuvchi servisni qayta yuklash kerak.</p> <ol style="list-style-type: none"> <li>1. etc/hosts va /etc/HOSTNAME fayllarining mazmunini o'zgartiring.</li> <li>2. Kompyuterni qayta ishga tushiring va DNS serverni yuklang.</li> </ol>

Agar laboratoriya ishini bajarish davomida xatoliklar paydo bo'lsa, quyida yuz bergan xatoliklar va ularning yechimlarini yozing:

---



---



---



---



---

## 12-laboratoriya ishi. Slave hududni sozlash

**Ishning maqsadi.** BIND serverida slave hududni qo'llash sozlamalarini o'rganish.

**Kiruvchi ma'lumotlar.** Slave hudud toq nomerli kompyuterlarda hosil qilinadi.

Topshirqlar	Mazmuni
1. named.conf fayliga hudud tavsifini qo'shish	1. Tahrirlash uchun /etc/named.conf faylini oching va unga hududni parametrlarini aniqlovchi zone ko'rsatmasini qo'shing. Quyida faylga kiritiladigan satrlarni yozing. <hr/> <hr/> <hr/>
2. Sintaksisni tekshirish	1. named-checkconf dasturi yordamida named.conf faylini sintaksis xatoliklarga tekshiring
3. DNS serverni ishga tushirish	1. DNS serverni yuklang named 2. Hudud haqidagi ma'lumotlarni saqlovchi fayl mazmunini qarab chiqing.

## 13-laboratoriya ishi. Teskari akslantirish hududini sozlash

**Ishning maqsadi.** BIND-da teskari akslantirish hududini qo‘llab-quvvatlashni amalga oshirishni o‘rganish

**Kiruvchi ma’lumotlar.** Teskari akslantirish hududi barcha juft kompyuterlarda yaratiladi (Sizning hushingiz DNS serverida joylashgan kompyuterlarda)

Teskari akslantirish hududida faqat uchta kompyuterni tavsiflash yetarli: Sizning domeningizga tegishli ikkita va o‘qituvchining kompyuteri.

Topshiriq	Mazmuni
1. /etc/named.conf fayliga hudud tavsifini qo‘shish	1. /etc/named.conf tahrirlash uchun oching va unga hudud parametrlarini aniqlovchi zone ko‘rsatmasini qo‘shing. Ushbu faylga kiritiladigan satrlarni yozing.  _____ _____ _____
2. Hududni tavsiflash faylini hosil qilish	1. Teskari akslantirish hududini tavsiflash faylini hosil qiling. Bu faylga joylashtirilgan ma’lumotlarni yozing.  _____ _____ _____
3. Sintaksisni tekshirish	1. named-checkconf dasturi yordamida named.conf faylini sintaksis xatoliklarga tekshiring. 2. named-checkzone dasturi yordamida teskari akslantirish hududi faylini sintaksisi xatoliklarga tekshiring.
4. Teskari akslantirish hududini qo‘llab quvvatlashni ishga tushirish	1. DNS serverni yangi hosil qilingan konfiguratsiya faylini o‘qishini ta’minlang killall -HUB named 2. /var/log/messages log faylining oxirgi satrlariga qanday ma’lumotlar qo‘shilganini tekshirib ko‘ring.
5. Ishga yaroqlilik (ish unumdorligi)	1. nslookup dasturini parametr sifatida o‘qituvchi kompyuterining IP adresini kiritgan holda ishga tushiring.

ni tekshirish	2. dig dasturini parametr sifatida –x parametri va o‘qituvchi kompyuterining IP adresini kiritgan holda ishga tushiring.
---------------	--

Agar laboratoriya ishini bajarish jarayonida xatoliklar paydo bo‘lsa, ularni yozib oling va xatoliklarni bartaraf etish usullarini yozing.

---

---

---

---



## 14-laboratoriya ishi. DNS-ni oddiy foydalanuvchi huquqi bilan ishlatish

### Ishning maqsadi:

BIND serverni oddiy foydalanuvchi huquqi bilan ishlatishni o'rganish.

Topshiriq	Mazmuni
1. Tayyorlash	1. /etc/rc.d/rc.bind faylini tahrirlash uchun ochish 2. DNS serverni ishga tushiruvchi funksiyaga –u parametrini qo‘shish /usr/sbin/named –u daemon 3. Faylni saqlash 4. Faylni bajariluvchi faylga aylantirish chmod a+x /etc/rc.d/rc.bind 5. Foydalanuvchi katologiga daemon-ni o‘tkazish chown –R daemon /var/named /var/run/named
2. Serverni ishga tushirish	1. DNS serverni qayta ishga tushiring /etc/rc.d/rc.bind restart 2. Adreslarni akslantirish to‘g‘ri bajarilayotganini tekshiring. Kompyuter sinfidagi barcha adreslarni tekshiring. Adreslarning to‘g‘ri akslantirilishi keyingi dars mashg‘ulotlari uchun zarur bo‘ladi.

## 15-laboratoriya ishi. Oddiy pochta serverini sozlash.

**Ishning maqsadi.** sendmail.cf konfiguratsiya faylini hosil qilishni o'rganish. Sendmail dasturini ishga tushirish va uning ishlashida paydo bo'ladigan muammolarni aniqlash.

**Kiruvchi ma'lumotlar.** Pochta serverini faqat toq o'rinlardagi kompyuterlarga sozlash kerak. Server bilan ishlash uchun pochta klientlari sifatida juft nomerdagi kompyuterlarni qo'llaymiz.

Topshiriqlar	Tavsifi
1. mc faylini yaratish	<ol style="list-style-type: none"><li>1. /usr/share/sendmail/cf/cf katologiga kiring</li><li>2. Ushbu katologda my.mc faylini yarating.</li><li>3. Faylning boshlang'ich satrlariga m4 preprotessoriga bog'lanish makroslarni kiriting: include ( `../m4/cf.m4` )dnl</li><li>4. Konfiguratsiya faylining versiyasini tavsiflovchi VERSIONID makrosini qo'shing: VERSIONID( `My personal cf` )dnl</li><li>5. Operatsion tizim turini tavsiflovchi OSTYPE makrosini qo'shing: OSTYPE(linux)dnl</li><li>6. local-host-names faylini qo'llash uchun bog'lash makrosi FEATURE ni qo'shing: FEATURE(use_cw_file)dnl</li><li>7. FEATURE makrosiga access.db kirish bazasini qo'llashni biriktiring FEATURE( `access_db`, `hash -T&lt;TMPF&gt; /etc/mail/access.db` )dnl Ushbu satr bitta satrda yozilishi kerak.</li><li>8. Ikkita MAILER makrosini qo'shing: MAILER(smtp)dnl MAILER(local)dnl</li><li>9. Faylni saqlang.</li></ol>
2. sendmail.cf faylini yaratish	<ol style="list-style-type: none"><li>1. sendmail.cf faylini yaratish uchun m4 preprotessoridan foydalaning. m4 my.mc &gt; /etc/mail/sendmail.cf</li><li>2. Faylning mazmunini 30-satridan boshlab ko'ring.</li></ol>

<p>3. Qo‘shimcha konfiguratsiya fayllarini tayyorlash</p>	<p>1. /etc/mail/local-host-names faylining oxiriga O‘zingizning domen nomini qo‘shing. Oxirgi satrni bo‘sh qoldirishni unutmang.</p> <p>2. /etc/mail/access fayliga qo‘shni kompyuterga pochta uzatishga ruxsat beruvchi satrni qo‘shing.</p> <p>Connect:IP_ADRES RELAY</p> <p>3. access.db faylini hosil qiling. /etc/mail katalogiga o‘ting va makemap dasturini bajaring.</p> <p>makemap hash access &lt; access</p>
<p>4. Dasturni yuklash</p>	<p>1. Barcha konfiguratsiya fayllarini tayyorlab bo‘lgandan keyin pochta tizimini qayta yuklang.</p> <p>/etc/rc.d/rc.sendmail restart</p> <p>2. Jarayonlar ro‘yxatida ikkita sendmail jarayoni borligiga ishonch hosil qiling.</p> <p>3. /var/log/maillog faylining mazmunini ko‘ring.</p>
<p>5. Foydalanuvchilarni qo‘shish.</p>	<p>1. Agar tizimda mavjud bo‘lmasa user1 foydalanuvchisini qo‘shing.</p> <p>2. Unga user1 parolini o‘rnating.</p>
<p>6. Pochta klientini sozlash</p>	<p>1. Juft nomerli kompyuterlarda kmail dasturini ishga tushiring.</p> <p>2. Nastroyka (sozlamalar) – Nastroit’ KMail... (KMail ni sozlash) men’yusini oching.</p> <p>3. Profil bo‘limini tanlang va Izmenit’ tugmasini bosing.</p> <p>4. Majburiy bo‘lgan maydon Adres e-mail ga o‘z pochta adresingizni kiriting. e-mail: user1@O‘z.domeningiz</p> <p>5. Vashe imya maydoniga ixtiyoriy nomni kiriting.</p> <p>6. OK tugmasini bosing.</p> <p>7. Otpravka jildidan – tarmoq bo‘limini tanglang.</p> <p>8. Ro‘yxatga yangi elementni qo‘shing. Agar u yerda qandaydir element bo‘lsa uni o‘chirish kerak. Qo‘shishda SMTP bandini tanlang.</p>

	<p>9. Imya maydoniga ixtiyoriy nomni kiritish mumkin. Server maydoniga albatta nom yoki jo‘natmalar yuboriladigan pochta serverining IP adresini kiritish kerak. Port maydoniga 25-portni kiriting. Qolgan maydonlarni to‘ldirish shart emas. OK tugmasini bosing.</p> <p>10. Poluchenie jildini tanlang.</p> <p>11. Yangi elementni qo‘shing. Qo‘shishda POP3 bandini tanglang.</p> <p>12. Imya maydonida ixtiyoriy nomni ko‘rsating. Uchetnoe imya maydoniga user1 deb yozing. Parol’ maydoniga user1 foydalanuvchining parolini kiriting. Server maydoniga pochta serverining nomini yoki IP adresini kiriting. Port maydoniga 110 ni kiriting. Xranit’ parol’ POP v fayle konfiguratsii bandini tanlang. OK tugmasini bosing.</p> <p>13. OK tugmasini bosing.</p> <p>14. Agar parollarni menedjerda saqlashga taklif paydo bo‘lsa, sozlash ustasiga maslahat qiling.</p>
<p>7. Jo‘natmalarni uzatish va qabul qilish</p>	<p>1. Serverdan jo‘natma qabul qilishga harakat qiling.</p> <p>Agar jo‘natmani qabul qilishda xatolik yuz bersa, o‘qituvchiga murojaat qiling.</p> <p>2. Dastlab jo‘natmani o‘zingizga yuborishga harakat qiling - user1@O‘z.domeningiz. Keyin sinfdagi boshqa domendan user1 foydalanuvchisiga pochta yuboring.</p> <p>Agar pochtani jo‘natishda xatolik yuz bersa, o‘qituvchiga murojaat qiling.</p>

Agar laboratoriya ishini bajarish davomida xatoliklar paydo bo‘lsa, uni yozib oling va ushbu xatolik oldini olish usullarini tavsiflab bering.

## 16-laboratoriya ishi. Virtusertable vositasini qo'llash

**Ishning maqsadi.** Virtusertable vositasini qo'llashni o'rganish.

Virtusertable yordamida virtual foydalanuvchilarning jo'natmalarini boshqa domendagi user1 foydalanuvchisiga uzatishni amalga oshiramiz. Domen nomlari virtual foydalanuvchilar nomiga bog'liq bo'ladi.

Topshiriqlar	Tavsifi
1. mc faylini o'zgartirish	<ol style="list-style-type: none"><li>1. /usr/share/sendmail/cf/cf katologiga o'ting.</li><li>2. my.mc faylini tahrirlash uchun oching.</li><li>3. MAILER makrosidan oldin quyidagi satrlarni qo'shing: FEATURE(`virtusertable', `hash -o /etc/mail/virtusertable.db')dnl Bu bitta satrga yozilishi kerak.</li><li>4. Faylni saqlang.</li><li>5. sendmail.cf faylini hosil qiling m4 my.mc &gt; /etc/mail/sendmail.cf</li><li>6. /etc/mail katologiga o'ting.</li><li>7. virtusertable fayliga quyidagi satrlarni qo'shing: virt1@u1 user1@ki.linux.zukko.uz virt2@u1 user1@di.linux.zukko.uz virt3@u1 user1@u3.linux.zukko.uz virt4@u1 user1@u4.linux.zukko.uz virt5@u1 user1@u5.linux.zukko.uz virt6@u1 user1@u6.linux.zukko.uz Tagiga chizilgan u1 domeni o'rniga o'ingizning domen nomini kiriting. Masalan: virt1@di.linux.zukko.uz</li><li>8. Faylni saqlang.</li><li>9. virtusertable.db hash jadvalini hosil qiling. makemap hash virtusertable &lt; virtusertable</li><li>10. Pochta serverini qayta yuklang /etc/rc.d/rc.sendmail restart</li></ol>
2. Ishchi holatni tekshirish	<ol style="list-style-type: none"><li>1. Ixtiyoriy virt foydalanuvchilaridan o'z domeningizga jo'natma yuboring.</li><li>2. Boshqa domendagi virt foydalanuvchisiga Domeningizga mos keluvchi nomer bilan</li></ol>

**17-laboratoriya ishi. Kirish kodidan dastur yig‘ish.**

**Ishning maqsadi:** Kiruvchi kodlardan squid proksi serverini yig‘ishni o‘rganish.

**Laboratoriya ishining topshirig‘i:** Kiruvchi kodlardan proksi serverni yig‘ing. Uni tizimga o‘rnatish.

<b>Topshiriqlar</b>	<b>Tavsifi</b>
1. Serverni yig‘ish	<p>1. O‘zingizning uy katologingizga proksi serverni arxivdan oching: tar -xjf squid-2.5.STABLE11.tar.bz2</p> <p>2. squid-2.5.STABLE11 katologiga kiring</p> <p>3. configure dasturini ishga tushiring: ./configure --prefix=/opt/squid \ --enable-delay-pools \ --enable-arp-acl \ --enable-default-err-language=Russian-koi8-r \ --enable-err-languages="Russian-koi8-r Russian-1251"</p> <p>Dastur xatolik haqidagi xabarsiz yakunlanishi shart. Agar xatolik paydo bo‘lsa – o‘qituvchiga murojaat qiling.</p> <p>4. make dasturini ishga tushiring. make</p> <p>5. Dasturni o‘rnatish: make install</p>
2. squid.sh fayli hosil qilish	<p>1. /etc/profile.d/squid.sh skriptini quyidagicha mazmunda yozing: #!/bin/bash PATH=\$PATH:/opt/squid/bin:/opt/squid/sbin export PATH</p> <p>2. Uni bajariluvchi faylga aylantiring: chmod a+x /etc/profile.d/squid.sh</p> <p>3. Tizimdan chiqib va yana qayta kiring.</p> <p>4. PATH o‘zgaruvchisining tarkibiga qarang. Uning tarkibida albatta quyidagi katologlar bo‘lishi shart: opt/squid/bin va /opt/squid/sbin</p>

## 18-laboratoriya ishi. Squid proksi serverini sozlash va ishga tushirish

**Ishning maqsadi.** Squid proksi serverini dastlabki sozlashni o‘rganish.

**Laboratoriya ishining topshirig‘i.** Proksi serverning dastlabki sozlamalarini bajaring. Klientlarga internetda \*.rambler.ru dan tashqari barcha resurslarga murojaat qilishga ruxsat bering.

Topshiriqlar	Tavsifi
1. squid.conf konfiguratsiya fayliga o‘zgartirishlar kiritish	<p>1. squid.conf faylini tahrirlash uchun oching.</p> <p>2. ACL tavsiflangan bo‘limga o‘ting.</p> <p>3. ACL-ning standart ro‘yxatidan keyin quyidagi satrlarni qo‘shing:</p> <pre>acl myhost src &lt;Sizning IP adresingiz&gt; acl mynet src &lt;Sizning tarmog‘ingiz IP adresi&gt; acl rambler dstdomain .rambler.ru acl wt time 10:00-17:00</pre> <p>Vaqtни o‘qituvchingiz bilan kelishib oling.</p> <p>4. http_access parametri tavsiflangan bo‘lim oxirida http_access_deny all –ning oxirgi satridan oldin quyidagi satrlarni qo‘shing:</p> <pre>http_access allow myhost http_access deny rambler http_access allow mynet wt http_access deny all</pre> <p>5. Faylni saqlang.</p>
2. Diskda keshni hosil qilish. Xatolarni tekshirish	<p>1. nobody foydalanuvchisiga /opt/squid katologini barcha tarkibi bilan bering:</p> <pre>chown -R nobody /opt/squid</pre> <p>2. squid-ni -z parametri bilan ishga tushiring:</p> <pre>squid -z</pre> <p>3. Agar bajarilish vaqtida ekranda xatolik ko‘rsatilsa, uni to‘g‘rilang va squid -z ni qayta ishga tushiring</p> <p>4. /opt/squid/var/cache katologini tarkibini qarab chiqing.</p> <p>5. Dasturni ishga tushiring:</p> <pre>squid</pre>

	6. Uchta jarayon: ikkita squid va bitta unlikd ishlayotganligiga ishonch hosil qiling ps xa
3. Klientlarni ulash	1. Brauzerni oching va uni qo'shni proksi serverlar bilan ishlashga sozlang. 2. www.rambler.ru sahifasiga kirishga urinib ko'ring. 3. Internetning ixtiyoriy boshqa resursiga kirishga harakat qiling. 4. /opt/squid/var/logs katologi tarkibini qarab chiqing.



## 19-laboratoriya ishi. Foydalanuvchilar autentifikatsiyasini sozlash

**Ishning maqsadi.** Foydalanuvchining autentifikatsiyasini sozlashni o'rganish.

**Laboratoriya ishining topshirig'i.** Proxy serverni \*.ya.ru va \*.yandex.ru resurslariga kirishda foydalanuvchilarni autentifikatsiyalashni talab qilishga sozlang.

Topshiriqlar	Tavsifi
1. Autentifikatsiya dasturlarini yig'ish	<p>1. Proksi serverning kiruvchi kodlar katologiga kiring.</p> <p>2. helpers/basic_auth/NCSA katologiga kiring.</p> <p>3. make dasturini ishlating.</p> <p>4. Dastur faylini nusxasini ko'chiring. cp ncsa_auth /opt/squid/sbin</p>
2. squid.conf konfiguratsiya fayliga o'zgartirish kiritish.	<p>1. squid.conf faylini tahrirlash uchun oching.</p> <p>2. Autentifikatsiyaning basic parametri aniqlangan satrni toping: auth_param basic children 5 auth_param basic realm Squid proxy-caching web server auth_param basic credentialsttl 2 hours</p> <p>3. Ushbu satrlardan oldin quyidagi ko'rinishda bitta satrni qo'shing: auth_param basic program /opt/squid/sbin/ncsa_auth /opt/squid/etc/passwd</p> <p>4. Oldingi laboratoriya ishidagi acl tavsifiga o'ting.</p> <p>5. acl ro'yxatiga http_access aniqlangan satrdan oldin quyidagi acl-ni qo'shing: acl passwd proxy_auth REQUIRED acl ya dstdomain .ya.ru .yandex.ru</p> <p>6. Quyidagi satrlar orasiga: <b>http_access deny rambler</b> <b>http_access allow mynet wt</b> <b>quyidagi satrlarni qo'shing:</b> <b>http_access allow ya passwd</b></p>

	<p><b>http_access deny ya</b></p> <p>7. Faylni saqlang.</p>
3. Foydalanuvchilar ro'yxati bilan faylni hosil qilish	<p>1. Quyidagi komandani bajaring: htpasswd -c /opt/squid/etc/passwd authuser</p> <p>2. authuser foydalanuvchisining parolini kiriting.</p> <p>3. /opt/squid/etc/passwd faylning mazmunini ko'rib chiqing.</p>
4. Proksi serverni qayta yuklash.	<p>1. Autentifikatsiya qo'shilgandan keyin proksi server o'zining konfiguratsiya faylini qayta o'qishini tashkil etish kerak: squid -k reconfigure</p>
5. Ishchi holatini tekshirish	<p>1. WEB brauzerni oching va www.ya.ru saytiga kirishga urinib ko'ring. Brauzer login va parolni so'rashi shart.</p> <p>2. authuser-ni va unga tayinlangan parolni kiriting.</p> <p>3. Ushbu brauzerda www.samtuit.uz sahifasini oching.</p> <p>4. Endi www.yandex.ru sahifasini oching. Autentifikatsiya uchun so'rov chiqmasligi kerak.</p> <p>5. Brauzerdan chiqing va uni qayta yuklang.</p> <p>6. www.ya.ru sahifasini oching. Avtorizatsiya qilish taklifi chiqishi kerak.</p>

## 20-laboratoriya ishi. Kanal kengligini cheklash sozlash.

**Ishning maqsadi.** Kanal kengligini cheklash sozlashni o'rganish.

**Laboratoriya ishining topshirig'i.** [www.zukko.uz](http://www.zukko.uz) sayti materiallarini ko'chirib olish tezligini sekundiga 40 Kilobaytgacha cheklang.

Topshiriq	Tavsifi
1. squid.conf konfiguratsiya fayliga o'zgartirishlar kiritish	1. squid.conf faylini tahrirlash uchun oching. 2. acl aniqlangan satrni toping va quyidagi satrlarni qo'shing: acl specialist dstdomain .zukko.uz 3. Fayl oxiriga sig'imni ushlab qolish tavsifini qo'shing: <b>delay_pools 1</b> <b>delay_class 1 1</b> <b>delay_parameters 1 400/400</b> <b>delay_access 1 allow</b> <b>specialist</b> <b>delay_access 1 deny all</b> 4. Faylni saqlang. 5. Proksi serverni o'zining konfiguratsiya faylini qayta o'qishini ta'minlang: squid -k reconfigure
2. Ishchi holatini tekshirish.	1. Brauzerda <a href="http://www.zukko.uz">www.zukko.uz</a> saytini oching. Ma'lumotlarni ko'chirib olish tezligiga e'tibor bering.

## 21-laboratoriya ishi. Limit mezonidan foydalanish.

**Ishning maqsadi.** Limit mezonidan foydalanishni o‘rganish.

**Topshiriq.** echo-request turidagi ICMP paketlarini sekundiga bir marta qabul qiladigan qilish kerak.

Topshiriqlar	Tavsifi
1. Ishni bajarishga tayyorgarlik ko‘rish.	<p>1. Sizning kompyuteringizda paketlarni filtrlashning qanday qoidasi o‘rnatilganligini tekshiring: iptables -L</p> <p>2. Agar qandaydir filtrlash qoidasi o‘rnatilgan bo‘lsa, uni o‘chirish kerak: iptables -F</p> <p>3. Agar siyosatda sukut bo‘yicha INPUT va OUTPUT zanjirlari ACCEPT qiymatlari o‘rnatilmagan bo‘lsa, sukut bo‘yicha siyosatga ACCEPT qiymatini o‘rnatish kerak. iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT</p>
2. INPUT zanjiriga qoidalarni qo‘shish	<p>1. Quyidagi qoidalarni qo‘shing: iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s --limit-burst 1 -j ACCEPT</p> <p>2. Bu qoidalar qoidalar ro‘yxatida qanday ko‘rinishini ko‘rib chiqing: iptables -L</p> <p>3. Bu qoidalar qanday ishlashini tekshiring. Qo‘shni kompyuterda quyidagi komandani bajarishni so‘rang: ping -f Sizning_IP -c 10000</p> <p>4. Yo‘qotilayotgan paketlar foiziga e‘tibor bering. U 0 yoki unga yaqin qiymatga teng, lekin bu qoidalar ishlamayapti degani emas. Qayta ishlanayotgan paketlar sonini qarang: iptables -L -v -x -n</p> <p>Ko‘rinib turibdiki, haqiqatan ham paketlar soni paketlarning umumiy sonidan kam. Limit moduli ishlov berilgan qoidalar sonini cheklaydi, agar qoidalarga ishlov berilmasa, zanjirda keyingi</p>

	qoidalar bajariladi. Keyingi qoida mavjud bo'lmaganda, sukut holatidagi ACCEPT siyosati bajariladi – paketlar qabul qilinadi.
3. Qo'shimcha qoidalarni kiritish	<p>1. Qabul qilinadigan paketlar sonini haqiqatan cheklash uchun yana bitta qoidani qo'shing:  iptables -A INPUT -p icmp --icmp-type echo-request -j DROP</p> <p>2. Bu qoida qanday ishlashini tekshiring. Qo'shni kompyuterda quyidagi komandani bajaring:  ping -f Sizning_IP -c 10000</p> <p>3. Yo'qotilayotgan paketlar soniga e'tiborni qarating.</p>

## 22-laboratoriya ishi. LOG amalidan foydalanish.

**Ishning maqsadi.** LOG amalidan foydalanishni o'rganish.

**Topshiriq.** Echo-request turidagi ICMP protokoli paketlari haqidagi ma'lumotlarni log tizimiga muhimlik darajasi debug-da joylashtirish zarur, lekin minutiga ikkita paketni bir martadan ortiq bo'lmagan holatda.

Topshiriqlar	Tavsifi
1. Ishni bajarishga tayyorgarlik	1. INPUT zanjirini barcha qoidalardan tozalang: iptables -F
2. Qoidalarni qo'shish (kiritish)	1. INPUT zanjiriga quyidagi qoidalarni kiriting: iptables -A INPUT -r icmp --icmp-type echo-request -m limit --limit 1/n --limit-burst 2 -j LOG --log-level debug --log-prolix "Alert: "
3. Ishchi holatini tekshirish	1. Qo'shni kompyuterdan quyidagi komandani bajaring: ping -s 10 SizningIP 2. icmp_seq maydoniga e'tibor bering, ketma-ketlikda xatoga yo'l qo'yilmayaptimi? Bu qoidalarga ishlov berilganliga qaramasdan paket o'z harakatini zanjir bo'ylab davom ettirishini bildiradi. 3. Ma'lumotlarni berilishini yoki syslogni quyidagicha sozlang, xabarlar log fayliga debug darajasida tushganligini ko'rish uchun dmesg dasturini bajaring.

## 23-laboratoriya ishi. NAT akslantirishni qo'llash

**Ishning maqsadi.** NAT akslantirishni qo'llashni o'rganish.

**Topshiriq.** Laboratoriya ishini ikki talaba bajarishga mo'ljalangan. Tarmoq ichida toq IP-adresli kompyuter klient sifatida ishlatiladi. Juft IP-adresli kompyuter NAT akslantirishni ishlab chiquvchi yo'naltiruvchi sifatida ishlatiladi.

Topshiriq	Tavsifi
1. Klient kompyuterida o'zgartirishlar kiritish	1. eth0 tarmoq interfeysini o'chiring: ifconfig eth0 down 2. Tarmoq interfeysiga kompyuteringizning eth1 interfeysi orqali ulangan routerning tarmoq interfeysiga standart marshrutni qo'shing: route add default gw 192.168.0.2
2. NAT akslantirishni yoqish.	1. Routerda interfeyslar orasidagi paketlarni yuborish imkoniyatini yoqing: echo 1 > /proc/sys/net/ipv4/ip_forward 2. Nat jadvaliga MASQUERADING amalini qo'shing: iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
3. Ishchi holatni tekshirish	1. Routerda iptraf dasturini ishga tushiring va barcha interfeyslarda paketlarni ko'rish rejimini yoqing. 2. Toq kompyuterlarda brauzerni ishga tushiring. 3. Proksi serverdagi brauzerni o'chiring. 4. Ixtiyoriy veb-sahifani oching. 5. Iptraf dasturidagi ulanishlarga qarang va ixtiyoriy ulanishda ikki nusxadagi ulanish ko'rsatilayotganiga e'tibor bering, bittasi eth1 interfeysida, ikkinchisi eth0 interfeysida.

## 24-laboratoriya ishi. NFS-dan foydalanish.

**Ishning maqsadi.** NFS-dan foydalanishni o'rganish.

**Topshiriq.** Sizning kompyuteringizda joylashgan /home/exports katologidan foydalanish uchun siz juft bo'lib ishlayotgan talabaga ruxsat bering. Qo'shni tomonidan eksport qilingan katologni mnt/nfs ulanish nuqtasiga ulang.

Topshiriq	Tavsifi
1. portmap-ni ishga tushirish	1. portmap tizimini ishga tushiruvchi skriptni bajariluvchi faylga aylantiring: chmod a+x /etc/rc.d/rc.portmap 2. /etc/hosts.allow fayliga quyidagi satrni qo'shing: portmap: ALL 3. Uni start parametri bilan ishga tushiring: /etc/rc.d/rc.portmap start 4. Xizmatning ishchi holatini tekshiring: rpcinfo -p localhost
2. Qayd yozuvi va katologlarni hosil qilish.	1. nfsnogroup guruhining mavjudligini tekshiring: grep nfsnogroup /etc/group 2. Agar bunday guruh bo'lmasa, uni hosil qiling: groupadd -g 65534 nfsnogroup 3. Tizimda nfsnobody foydalanuvchisi mavjudligini tekshiring: id nfsnobody 4. Agar bunday foydalanuvchi bo'lmasa, uni hosil qiling: useradd -u 65534 -g nfsnogroup -s /bin/false nfsnobody 5. Tarmoq bo'yicha taqsimlanadigan katologni hosil qiling: mkdir -m 777 /home/exports 6. Qo'shni kompyuterdagi tarmoq fayl tizimiga ulaydigan katologni hosil qiling: mkdir mnt/nfs
3. NFS-serverini sozlash	1. Quyidagi satrni /etc/exports fayliga qo'shing: /home/exports *(rw)



	<p>2. NFS-serverini ishga tushiruvchi skriptni bajariluvchi faylga aylantiring:  <code>chmod a+x /etc/rc.d/rc.nfsd</code></p> <p>3. NFS-serverini ishga tushiring:  <code>/etc/rc.d/rc.nfsd start</code></p> <p>4. Quyidagi satrlarni <code>/etc/hosts.allow</code> fayliga qo'shing:  <code>lockd:ALL</code>  <code>mountd:ALL</code>  <code>rquotad:ALL</code>  <code>statd:ALL</code></p> <p>5. Kompyuteringizda RPC protseduralari ro'yxatini ko'rib chiqing:  <code>rpcinfo -p localhost</code></p>
<p>4. Tarmoq diskiga ulanish</p>	<p>1. Qo'shni kompyuter tomonidan taqdim etilgan RPC protseduralari ro'yxatini ko'rib chiqing:  <code>rpcinfo -p qo'shni-kompyuter-IP-adresi</code></p> <p>2. Tarmoq diskiga ulaning:  <code>mount -t nfs IP:/home/exports /mnt/nfs</code></p> <p>3. Ulangan disklar ro'yxatini ko'rib chiqing:  <code>mount</code></p> <p>4. <code>/mnt/nfs</code> katologida fayl hosil qiling va qo'shni kompyuterdan hosil qilingan faylga kirish huquqi ko'rishini so'rang.</p> <p>5. Tarmoq diskini o'chiring:  <code>umount /mnt/nfs</code></p>

## 25-laboratoriya ishi. Virtual xostingni sozlash.

**Ishning maqsadi.** Apache-serverida WEB-serverlarning virtual xostingini sozlashni o'rganish.

**Topshiriq.** [www.samtuit.uz](http://www.samtuit.uz) saytining virtual xostingini sozlang.

Topshiriq	Tavsifi
1. Foydalanuvchini qo'shish	<p>1. any foydalanuvchisini any paroli bilan qo'shing. useradd -m -g nobody -s /bin/false any passwd any</p> <p>2. public_html katologini hosil qiling: mkdir /home/any/public_html chown any:nobody /home/any/public_html chmod 750 /home/any/public_html</p> <p>3. Server loglarni joylashtiradigan katologni hosil qiling: mkdir /home/any/err chown any:nobody /home/any/err chmod 770 /home/any/err</p> <p>4. public_html katologida index.html faylini quyidagi mazmunda hosil qiling: &lt;HTML&gt; &lt;HEAD&gt; &lt;TITLE&gt; TEST FILE &lt;/TITLE&gt; &lt;/HEAD&gt; &lt;BODY&gt; test &lt;/BODY&gt; &lt;/HTML&gt;</p>
2. Virtual xostingni sozlash	<p>1. Apache WEB-serverining konfiguratsiya faylini tahrirlash uchun oching: mcedit /etc/apache/httpd.conf</p> <p>2. NameVirtualHost parametridan komentariya belgilarini olib tashlang: NameVirtualHost *:80</p> <p>3. O'z serveringiz va virtual xostingning tavsifini qo'shing: &lt;VirtualHost *:80&gt; ServerAdmin root@localhost</p>

	<pre> DocumentRoot /var/www/htdocs ServerName c1.ki.linux.zukko.uz ErrorLog var/log/apache/error_log CustomLog      /var/log/apache/access_log common &lt;/VirtualHost&gt;  &lt;VirtualHost *:80&gt; ServerAdmin any@samduit.uz DocumentRoot /home/any/public_html ServerName www.samduit.uz ErrorLog /home/any/error_log CustomLog      /home/any/err/access_log common &lt;/VirtualHost&gt; 5. Faylni saqlang. 6. Konfiguratsiya faylini xatoliklar mavjudligiga tekshiring: apachectl configtest </pre>
<p>3. Serverni ishga tushirish</p>	<p>1. Serverni ishga tushiruvchi skriptni bajariluvchi faylga aylantiring:  <code>chmod a+x /etc/rc.d/rc.httpd</code></p> <p>2. Serverni ishga tushiring:  <code>/etc/rc.d/rc.httpd start</code></p> <p>3. <code>/etc/hosts</code> faylida kompyuteringiz IP-adresi satriga <code>www.samduit.uz</code> nomini kiriting.  Haqiqiy xositingni sozlashda bu qadamni bajarish shart emas.</p>
<p>4. Ishchi holatni tekshirish.</p>	<p>1. Brauzerni oching va kompyuteringizning FQDN nomini kiriting. Masalan:  <code>http://c1.ki.linux.zukko.uz</code>  Natijada eski saytning (Apache bo'yicha hujjat) mazmunini olasiz.</p> <p>2. Brauzerda URL-ga: <code>www.samduit.uz</code> nomini kiriting. Natijada <code>www.samduit.uz</code> saytining test sahifasi ochiladi.</p>

## 26-laboratoriya ishi. FTP-serverni sozlash.

**Ishning maqsadi.** VSFTPD FTP-serverini anonim kirish rejimida sozlashni o‘rganish. FTP-serverga lokal foydalanuvchilarning kirishini cheklashni o‘rganish.

**Topshiriq.** O‘zingizning WEB-saytingiz materiallariga foydalanuvchilarning kirishini tashkil eting. Shuningdek, ommaviy FTP-serverni sozlang, unga anonim foydalanuvchilar fayllarni ko‘chirib olish imkoniyatini bering.

Topshiriqlar	Tavsifi
1. any foydalanuvchisiga uy katalogiga ftp-server orqali kirishga ruxsat berish.	<p>1. FTP-serverning /etc/vsftpd.conf – konfiguratsiya faylini tahrirlash uchun oching.</p> <p>2. local_enable parametriga YES qiymati ta'minlanganligiga ishonch hosil qiling</p> <p>3. write_enable parametriga YES qiymati ta'minlanganligiga ishonch hosil qiling</p> <p>4. Faylni yoping.</p>
2. Foydalanuvchilarning tizimga kirishini cheklash.	<p>Foydalanuvchilarning uy katalogiga kirishga ruxsatni faqat ftp-serveri orqali amalga oshiring.</p> <p>1. /etc/shells faylini tahrirlash uchun oching va quyidagi satrni qo‘shing: /bin/false</p> <p>2. Boshqa virtual qobiq orqali tizimga any foydalanuvchisi sifatida kirishga harakat qilib ko‘ring.</p> <p>3. /etc/hosts.allow fayliga quyidagi satrni qo‘shing: vsftpd: ALL</p> <p>4. Ixtiyoriy WEB-brauzerda quyidagi URL-ni kiriting: ftp://any@www.samtuit.uz</p>
3. Anonim kirishni tashkil etish.	<p>1. FTP-serverning /etc/vsftpd.conf faylini tahrirlash uchun oching.</p> <p>2. anonymous_enable parametri YES qiymatiga ega ekanligiga ishonch hosil qiling.</p> <p>3. anon_upload_enable parametriga YES qiymatini ta'minlang</p>

	<p>4. anon_mkdir_write_enable parametriga YES qiymatini ta'minlang</p> <p>5. /home/ftp/incoming katologini hosil qiling va uni ftp-foydalanuvchi va ftp-guruhiga bering.</p> <pre>mkdir /home/ftp/incoming chown ftp:ftp /home/ftp/incoming</pre> <p>6. /home/ftp/pub katologini hosil qiling:</p> <pre>mkdir /home/ftp/pub chown ftp:ftp //home/ftp/pub</pre> <p>7. pub-katologiga qandaydir faylni joylashtiring. Bu fayl ftp-foydalanuvchilari uchun o'qishga ruxsat berilgan bo'lishi shart.</p> <p>8. FTP-serverga kirish uchun mc-dasturidan foydalaning. pub-katologining mazmuniga qarang. Qandaydir faylni incoming katologiga o'tkazing.</p> <p>9. incoming katologida fayl kimga tegishli ekanligini ko'ring.</p>
--	--

## FOYDALANILGAN ADABIYOTLAR

1. Sobell, Mark G. A practical guide to Linux commands, editors, and shell programming / Mark G. Sobell. -2nd ed. p. cm. Includes bibliographical references and index. ISBN 978-0-13-136736-4 (pbk.) P 1081.
2. Крюков А.А. Linux учебное пособие по курсу: Расширенное администрирование OS UNIX (Linux). Москва, 2006. <http://www.specialist.ru>.
3. Computer networking : a top-down approach / James F. Kurose, Keith W. Ross. -6th ed., Pearson, 2013.
4. Иванов Н. Н. Программирование в Linux. Самоучитель. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2012. - 400 с. ISBN 978-5-9775-0744-8
5. Колисниченко Д.Н. Руководство по командам и shell-программированию в Linux. - СПб.: БХВ-Петербург, 2011. - 288 с. ISBN 978-5-9775-0619-9
6. Колисниченко Д. Н. Серверное применение Linux. - 3-е изд., перераб и доп. - СПб.: БХВ-Петербург, 2011. - 528 с. - (Системный администратор) ISBN 978-5-9775-0652-6
7. Старовойтов А. А. Сеть на Linux: проектирование, прокладка, эксплуатация. - СПб.: БХВ-Петербург, 2005. - 288 с.: ISBN 5-94157-687-0

## GLOSSARIY

<b>absolute pathname</b>	Absolyut (mutloq) yo'l nomi. Faylga yo'l ildiz katalog nomi va qismkatologlar nomi orqali ko'rsatilishi (yo'l / belgisi bilan korsatiladi, bu belgi katologlar, qismkatologlar hamda fayl nomlarini ajratish uchun qo'llaniladi). Masalan, /usr/bin/bash. Absolyut yo'l faylni joriy katologga bog'liq bo'lmagan holda topishga yordam beradi.
<b>access</b>	Kirishga ruxsat. Bu kompyuter jargoni bo'lib, o'qish yoki yozish uchun ma'nolarini anglatadigan so'z. Faylga o'qish yoki yozish uchun kirishni anglatadi
<b>Access Control List</b>	ACL – ga qarang.
<b>access permissions</b>	Faylda o'qish, unga yozish yoki uni bajarish uchun ruxsat. Agar faylga yozish uchun ruxsat berilgan (odatda, faqat yozish ruxsati) bo'lsa, faylga ma'lumot yozishingiz mumkin. Shuningdek, yozish imtiyoziga ega bo'lasiz.
<b>ACL</b>	Access Control List (Kirishga nazorati ro'yxati). Faylga kirish ruxsatiga o'xshash vazifani bajaradigan, lekin juda nozik nazorat bilan ishlaydigan tizim
<b>active window</b>	Faol oyna - ish stoli, klaviaturada yozgan belgilarni qabul qiluvchi oyna
<b>address mask alias</b>	Qism tarmoq maskasiga qarang Yangi buyruqlarni tavsiflash (belgilash) imkonini beruvchi shell (qobiq)da qo'llaniladigan mexanizmi
<b>API</b>	Application program interface (Amaliy (ilova) dastur interfeysi). Ilova dasturining operatsion tizim va uning boshqa xizmatlarga kirish interfeysi (chaqiruv konventsiyalari). API kirish kodi darajasida aniqlanadi (belgilanadi) va kodning mavhumligini ta'minlash uchun dastur va yadro (yoki boshqa imtiyozli utilitlar) o'rtasida hamkorlik (moslashuvchanlik) darajasini aniqlab beradi.
<b>applet</b>	Kattaroq dastur doirasida ishlaydigan kichik dastur.

	Masalan, brauzer va ish stol ilovalar panelida ishlaydigan Java dasturlari (ilovalari).
<b>archive</b>	Biriga bog‘liq bo‘lgan fayllarguruhini o‘z ichiga olgan kichik hajmli fayl. Jumladan, shu turdagi fayllarni yaratuvchi dasturlar.
<b>array</b>	Bir yoki bir nechta o'lchamdagi elementlar (satrdagi belgilar soni yoki belgilar majmuasi) majmuasi.
<b>ASP</b>	Application Service Provider (Ilovalarga xizmat ko‘rasuvchi provayder). Internet orqali ilovalarni ta'minlovchi kompaniya
<b>Berkeley UNIX</b>	UNIX operatsion tizimining ikkita asosiy versiyalaridan biri. Berkeley UNIX kompyuter tizimlari tadqiqot guruhi tomonidan Berkeleydagi Kaliforniya Universitetida ishlab chiqilgan va odatda BSD (Berkeley Software Distribution) deb nomlanadi.
<b>BIND</b>	Berkeley Internet Name Domain (Berkeley Internet Domeni Nomi). Berkeleydagi Kaliforniya Universiteti tomonidan ishlab chiqilgan va tarqatilgan DNS serverni joriy qiluvchi tizim.
<b>BIOS</b>	Shaxsiy kompyuterlarda tashqi qurilmalar uchun eng quyi darajadagi (standart) interfeysni ta'minlaydi. Operatsion tizimlarning yuklovchi (boot) jarayonini ishga tushiradi. BIOS doimiy elektr bilan ta'minlanuvchi xotira - BIOS ROM-da saqlanadi. Shuning uchun ham kompyuter o‘chirilganda ham barcha sozlamalar saqlanib qoladi.
<b>bit</b>	Kompyuterdagi ma'lumotning eng kichik qismi. Bit – ikkilik (binar) raqam: 1 yoki 0 (bor yoki yo‘q – rost yoki yolg‘on).
<b>block device</b>	Blokli qurilma - Disk yoki lenta drayveri. Blokli qurilma ma'lumotlarni belgilar blokida saqlaydi. Blokli qurilma blokli qurilma fayli bilan ifodalanadi.
<b>Bourne Again Shell</b>	bash. UNIX uchun GNU komandalari interpretatori. Bash-lar - Bourne Shell-ning to'liq sintaksisiga mos keluvchi C Shell-ga o‘rnatilgan



komandalari bilan hamkorlikda ishlovchi interpretator hisoblanadi. Bourne Again Shell emacs-matn redaktori stilidagi komandalar satrida tahrirlashni qo'llab-quvvatlaydi, topshiriqlar, funksiyalar va interfaol ma'lumotnomalarni boshqaradi.

### **Bourne Shell**

sh. Bu AT&T Bell Laboratories-da Stiv Bourne tomonidan ishlab chiqilgan UNIX buyruqlar protsessori.

### **BSD**

Berkeley UNIX-ga qarang.

### **buffer**

Vaqtinchalik xotira. Ma'lumotlarni undan foydalanmaguncha bo'lgan vaqt mobaynda saqlab turuvchi xotira sohasi. Diskdagi faylga ma'lumotlarni yozayotganda diskka yozish uchun yetarli miqdorda ma'lumot to'planmaguncha yoki disk ma'lumotlarni qabul qilishga tayyor bo'lmaguncha Linux ma'lumotlarni buferda saqlab turadi.

### **byte**

Bayt. Kompyuter ma'lumotlari ierarxiyasidagi bitdan katta, so'zdan kichik bo'lgan komponent (o'lchov birligi). Hozirgi kunda sakkiz bit va eng kichik adreslanadigan ma'lumot birligi. Odatda bayt bitta belgini o'zida saqlaydi. Komponent v ierarxii mashinnyx dannyx, obychno bol'she, chem bit i men'she, chem slovo; seychas chaste vsego vosem' bit i naimen'shaya adresuemaya yedinitsa xraneniya. Bayt obychno soderjit odin simvol

### **C programming language**

C dasturlash tili. Samarali modulli dasturlash uchun yuqori darajali funksiyalarni tadbiq etuvchi, shuningdek, tizimli dasturlarni yozish uchun quyi darajali funksilar bilan boyitilgan zamonaviy dasturlash tili. Linux-ning asosiy qismi C tilida yozilgan, shuning uchun ham Linux C tilida dastur yozish uchun ideal muhitni taqdim etadi.

### **C Shell**

csh. Bu C Shell buyruqlar protsessori Bill Joy tomonidan BSD UNIX uchun ishlab chiqilgan. Bunga C dasturlash tilining nomi berilganligining sababi bu tildagi dasturlash

	konstruksiyalari C tilidagiga o'xshash.
<b>cache</b>	Kesh - o'ta tezkor kichik xotira bo'lib, oxirgi foydalanigan ma'lumotlarga qayta murojaatni tezlashtirish uchun mo'ljallangan.
<b>character device</b>	Belgili qurilmalar: terminal, printer yoki modem qurilmalari. Bunday qurilmalar bir vaqtning o'zida belgilarni saqlaydi va ko'rsatib turadi. Bu qurilmalar maxsus belgili fayl bilan ifodalanadi. Blokli qurilmalar bilan aynan bir xil.
<b>checksum</b>	Nazorat summasi. Bu ma'lumotlar blokining mazmuniga bog'liq bo'lgan ma'lumotlar yig'indisini aniqlash uchun ma'lumotlar bilan birga uzatiladigan yoki saqlanadigan qiymat. Qabul qiluvchi tizim olingan ma'lumotlarga asoslanib nazorat summasini qayta hisoblab chiqadi va natijani ma'lumot bilan birga yuborilgan qiymat bilan solishtiradi. Ikki qiymat bir xil bo'lsa, qabul qiluvchining ishonchli ma'lumotni to'g'ri qabul qilganligi tasdiqlanadi. Bu miqdor 8, 16 va 32 bitga yoki boshqa hajmdagi o'lchovga teng bo'lishi mumkin. Bunda ma'lumot blokidagi baytlar yoki so'zlar yig'indisi hisoblab chiqiladi, bu yerda overflow hisobga olinmaydi. Bu qiymat inkor qilinishi mumkin, shuning uchun ham bu qiymat nolga teng deb olinadi. Internetda nazorat summasi 32 bitli ma'lumot paketidan foydalaniladi.
<b>child process</b>	Boshqa jarayon ("ota" jarayon)lar tomonidan yaratiladigan jarayon. Linux-ni ishga tushiruvchi birinchi jarayondan tashqari barcha har qanday jarayon child ("bola") jarayon hisoblanadi. Shell dan buyruqni bajarishga yuborganimizda, shell ushbu buyruqni bajarish uchun child jarayonni yaratadi.
<b>CIDR</b>	Classless Inter-Domain Routing (Sinfsiz Domenlararo Marshrutlash). Internet adreslari bloklarini taqsimlovchi sxema bo'lib, ularni marshrutlash jadvalidagi kam sonli yozuvlar bilan birlashtirish imkonini beradi. CIDR bloki – bu

Internic tomonidan ISP-lar uchun tayinlanadigan internet adreslar bloki.

<b>CIFS</b>	Common Internet File System (Umumiy Internet Fayl tizimi). SMB-ga asoslangan internet fayl tizimi protokoli. CIFS - TCP/IP-ning ustki qismida ishlaydi, DNS-dan foydalanadi va sekin ishlovchi dial-up Internet-ulanishlarini qo'llab-quvvatlash uchun optimallashtirilgan. SMB va CIFS bir-birining ornida qo'llaniladi.
<b>CIPE</b>	Crypto IP Encapsulation (Kriptografik IP-himoyasi). Bu protokolning shifrlangan UDP-paketlaridagi IP-paketlari juda oddiy bo'lib, dinamik manzillar, NAT va SOCKS proksi-serverlari ustida ishlaydi.
<b>Classless Inter-Domain Routing CLI</b>	CIDR-ga qarang
<b>client</b>	Command-line interface (Kommandalar satri interfeysi). Shuningdek, belgili yoki matnli interfeys deb ham yiritiladi.
<b>command line</b>	Serverdan bir yoki bir necha xizmatlarni talab qiluvchi kompyuter yoki dastur
<b>command substitution</b>	Foydalanuvchi tomonidan berilgan buyruqni bajarish uchun ko'rsatmalar va parametrlarni o'z ichiga olgan satr.
<b>console</b>	Buyruqni chiqishi bilan almashtirish. Agar komandani \$ belgisi yoki teskari apostrof belgilari (^ `) juftligi orasiga olib yozilsa, qobiq dasturi komandani o'rniga qo'yishni bajaradi.
<b>CPU</b>	Tizimning asosiy terminali – foydalanuvchi bilan muloqot uchun buyruqlar satri. Shuningdek, tizimli konsol yoki terminal konsoli deb ham ataladi.
	Central processing unit (markaziy protsessor bloki). Kompyuterning boshqa barcha qismlarini nazorat qiluvchi asosiy qism. CPU boshqaruvni va arifmetik - mantiqiy hisoblashlarni (ALU) o'z ichiga oladi. Boshqarish bloki ko'rsatmalarni xotiradan oladi va ularni kompyuterning boshqa qismlarini nazorat qilish signallarni ishlab chiqish

	uchun taqsimlaydi.
<b>daemon</b>	Oshkor ravishda chaqirilmaydigan dastur, lekin qandaydir shart bajarilishini (hodisa ro‘y berishini) kutish holatida bo‘ladi. Bu shartni bajaruvchi tomonidan demon (dasturi) yashirin holatda ekanligini bilishga majbur emas (ba‘zi hollarda dastur faqat bir ishni amalga oshirishi mumkinligini bilishi mumkin, chunki u demonni bajaradi). Bu atama mifologik jihatidan Disk And Execution MONitor (Disk va ijro etuvchi monitor) so‘zlarining qisqartmasi sifatida tadbiiq qilingan.
<b>data structure</b>	Ma‘lumotlar tuzilmasi. Ma‘lumotlarni saqlash, tashkillashtirish, qayta ishlash va olish uchun qo‘llaniladigan maxsus format. Ma‘lumotlar tuzilmasi – ma‘lumot elementlari va ular orasidagi munosabatlar bilan aniqlanadi. Misollar: bog‘langan ro‘yxatlar, daraxt, fayllar, yozuvlar, jadvallar, massivlar.
<b>datagram</b>	Tarmoqda ma‘lumotlarni almashish uchun qismlarga ajratiladi. Alohida qism deytagram deyiladi. Ya‘ni deytagram uzatuvchi kompyuteridan qabul qiluvchi komp’yuteriga jo‘natmani marshrutlash uchun yetarlicha ma‘lumotlarga ega bo‘lgan mustaqil ma‘lumot ob‘ekti hisoblanadi, bunda marshrutlashda ma‘lumotlarni oldingi almashinuvlari e‘tiborga olinmaydi. UDP – protokoli deytagramdan foydalanadi. IP – protokollar esa paketlardan foydalanadi. Paketlar tarmoq darajasida (sathida) bo‘linmas birlikdagi ma‘lumotning bir qismi.
<b>debug</b>	Dasturni xatolardan halos qilish orqali tuzatish maqsadida kompiliatsiya qilish
<b>default</b>	Oldindan kelishilgan holat yoki standart sifatida qabul qilingan holat. Masalan, biror komanda zarur parametrsiz kiritilganda sukut bo‘yicha holat tanlanishi. Jimlik (sukut) bo‘yicha qabul qilinadigan holatlar.
<b>Desktop</b>	Foydalanuvchiga qulaylik yaratish uchun taklif

etiladigan ishchi oyna. Misol uchun, bir yoki bir nechta piktogrammalar majmuasi, instrumentlar paneli va h.k.

<b>device</b>	Kompyuterga ulanishi mumkin bo'lgan qurilma. Masalan, disk qurilmasi, printer, terminallar, plotter va boshqa kiritish-chiqarish qurilmalari. Tashqi qurilamalarning qisqa nomi.
<b>device driver</b>	Terminal, disk yoki printer kabi qurilmani boshqaradigan Linux yadrosining bir qismi
<b>device file</b>	Qurilmani tavsiflovchi fayl.
<b>device number</b>	Asosiy (major device) qurilma va kichik minor device) qurilmani anglatuvchi son
<b>DHCP</b>	Dynamic Host Configuration Protocol (Dinamik Xost Konfiguratsiya Protokoli). LAN (lokal tarmoq)dagi kompyuterlarga IP adreslarni dinamik (o'zgaruvchan) taqsimlovchi protocol.
<b>directory</b>	Fayllarni birlashtiruvchi katolog.
<b>distributed computing</b>	Taqsimlangan hisoblash. Qandaydir masala yoki xizmatlarni korporativ tizimlar tarmogida taqsimlash orqali yechimini olish usuli, bunday tizimlarning ko'pchiligi ixtisoslashgan bo'lishi mumkin
<b>DMZ</b>	Demilitarized zone (Egallangan hudud). Lokal tarmoq va Internet orasidagi neytral hududga ega bo'lgan tarmoq tuguni yoki kichik tarmoq. U veb-sahifalar va Internetning boshqa ma'lumotlariga xizmat qilishi mumkin va lokal tizimlarning Internetga kirishiga imkon yaratadi, lokal tarmoqdagi Internetda ro'yxatdan o'tmagan foydalanuvchilarning kirishini taqiqlaydi. Hatto agar DMZ buzilgan bo'lsa ham, u hech qanday konfidentsial ma'lumotlarni saqlab qolmaydi.
<b>DNS</b>	Domain Name Service (Domen nomi xizmati). To'liq xost (domen nomi qo'shilgan) nomining IP-adreslar va tizimning boshqa xususiyatlarining mosligini boshqaruvchi taqsimlangan xizmat.
<b>DOM</b>	Document Object Model (Hujjatning obyekt modeli). Dasturning mazmunini, tuzilishini va uslubini dinamik ravishda yangilashga imkon

beruvchi platforma / tildan mustaqil interfeys.

O'zgarishlar keyinchalik ko'rsatilgan hujjatning bir qismiga aylanishi mumkin. Qo'shimcha ma'lumot uchun [www.w3.org/DOM](http://www.w3.org/DOM) sahifasiga kirib ko'ring.

**DSA** Digital Signature Algorithm (Raqamli imzo algoritmi). Raqamli imzolarni yaratish uchun ochiq kalitli shifrlash ishlatilgan.

**DSL** Digital Subscriber Line/Loop (Raqamli Abonent tarmog'i / Loop). Maxsus, shartli telefon tarmog'i bo'yicha tezkor aloqa bilan ta'minlaydi.

**Ethernet** Bir soniyada 1000 Megabitgacha yuqori tezlikda uzatish imkoniyatini beruvchi LAN turi.

**FDDI** Fiber Distributed Data Interface (ma'lumotlarni uzatishning shisha tolali interfeysi). Ma'lumotlarni shisha tolali kabellar bo'yicha sekundiga 100 million bit tezlik bilan uzatish uchun mo'ljallangan lokal tarmoq turi.

**file** Fayl - diskning nomlangan qismi. Fayllar o'zida saqlayotgan ma'lumotlar toifasiga qarab bir necha turlarda bolishi mumkin. Masalan, eslatmalar, hisobotlar, xabarlar, dasturlash tilidagi kodlar, ro'yxatlarni saqlovchi fayllar matnli fayl bo'ladi. Bajariladigan yoki ishga tushiriladigan fayllar binary fayllar yoki dastur fayllari bo'ladi.

**filesystem** Fayl tizimi – Odatda diskning bir qismida joylashgan ma'lumotlar tuzilmasi. Barcha Linux tizimlari ildiz fayl tizimiga ega, ko'pchilik tizimlarda esa boshqa fayl tizimlari mavjud. Har bir fayl tizimi fayl tizimiga mo'ljallangan disk bo'limining o'lchamiga bog'liq bo'lmagan bir nechta sondagi bloklardan tashkil topadi. Har bir fayl tizimi superblok deb ataluvchi boshqaruv blokiga ega, bu blok fayl tizimi haqidagi ma'lumotlarni saqlaydi. Fayl tizimidagi boshqa bloklar fayllar haqidagi boshqarish ma'lumotlari va fayldagi ma'lumotlarni saqlovchi tugunlarni o'z ichiga oladi.

**filter** Standart kiritishda kiritiladigan va uni chiqarishda standart chiqariladigan komanda. Filtr kiruvchi

ma'lumotlar oqimini akslantiradi va uni standart chiqishga jo'natadi. O'tkazish kanali (pipe) filtrni kirish bilan bog'laydi, ikkinchi kanal esa standart chiqish bilan bog'laydi. Grep va sort utilitlari odatda filtr sifatida qo'llaniladi.

## **firewall**

Tarmoq xavfsizligini ta'minlash uchun foydalaniladigan siyosatga asoslangan tarmoq trafiginı boshqarish qurilmasi. Brandmauer (firewall) keraksiz paketlarni filtrlashga mo'ljallangan bitta marshrutizatorga qo'llanilgan bo'lishi mumkin, yoki u marshrutizatorlar, proksi serverlar va boshqa qurilmalar kombinatsiyasiga joylashtirilishi mumkin. Firewall foydalanuvchilarning Internetga xavfsiz kirishi uchun va kompaniyalarning WWW-serverlarida ularning ichki tarmog'idan umumiy kirishlarni taqsimlash uchun keng qo'llaniladi. Shuningdek tarmoqning ichki segmenti xavfsizligini ta'minlash uchun ham qo'llaniladi. So'nggi paytlarda ushbu atama mustaqil aniqlanmoqda va oxirgi nuqtada ishlovchi komp'yuterda paketlarni oddiy filtrlash kiritilgan.

## **FQDN**

Fully qualified domain name (To'liq aniqlangan domen nomi). Tizimning to'liq nomi xost nomi va yuqori sathdagi domen qo'shilgan domen nomidan tashkil topgan. Masalan, resource – bu xost nomi, resource.zukko.uz – bu to'liq domen nomi. Internetda komp'yuterning unikal internet adresini aniqlash uchun FQDN yetarli bo'ladi.

## **gateway**

Shlyuz. O'zaro ma'lumotlarni almashish uchun bir nechta turli muhitli tarmoqqa ulangan komp'yuterlar yoki maxsus qurilmalarni belgilovchi umumiy atama. Shlyuzlar ko'pincha marshrutizatorlardan farqli ravishda ma'lumotlarni uzatishdan oldin ularni boshqa formatga o'zgartirib oladi. Shlyuzlardan foydalanish hozirgi kunda bekor qilinmoqda.

## **group (of users)**

Foydalanuvchilar guruhi. Guruhdan tarmoqda fayllardan foydalanishga umumiy ruxsat berish

	uchun qo‘llaniladi.
<b>group ID</b>	Foydalanuvchilar guruhini aniqlovchi unikal raqam. Bu parol va guruhlar haqidagi ma’lumotlar bazasida saqlanadi (/etc/passwdand, /etc/groupfiles yoki ularning ekvivalenti NIS). Bu guruhlar ma’lumotlar bazasida guruh ID-si gurih nomi bilan bog‘lanadi. Bu odatda GID deb ham yuritiladi.
<b>GUI</b>	Graphical user interface (Foydalanuvchining grafik interfeysi). Operatsion tizimning belgilar (piktogramlar)ga asoslangan grafik ob’ektlilik interfeysi.
<b>home directory</b>	Uy katologi. Birinchi kirgan ishchi katalog. Bu katalogga yo‘l HOME shell o‘zgaruvchisida saqlanadi.
<b>HTML</b>	Hypertext Markup Language (Giper matli belgilashlar tili). World Wide Web da foydalaniladigan gipermatli hujjat formatil. Teglar “kichik” (<) belgisi, nol va boshqa parametrlar va “katta” (>) belgisidan iborat. <TITLE> va </TITLE> kabi teglar juftligi matn yoki hujjatning aniq bir stilini aniqlash uchun qo‘llaniladi.
<b>HTTP</b>	Hypertext Transfer Protocol (Gipermatnli uzatish protokoli). HTML-hujjatlarni almashish uchun World Wide Web-da foydalaniladigan kliyent / serverning TCP / IP protokoli.
<b>ICMP</b>	Internet Control Message Protocol (Internetda xabarlarni boshqarish protokoli). Faqat ma’lumotlar haqida xabarlar (ma’lumotning o‘zi bo‘lmaydi)ni olib yuruvchi tarmoq paketi
<b>icon</b>	Foydalanuvchining grafik interfeysi (GUI)da katalog, fayl, dastur yoki biror komandani aks ettiruvchi kichik rasm (piktogramma). Bu tarixiy minaturalardagi diniy haykal (ikona)lardan olingan.
<b>interface</b>	Ikki qismtizimning uchrashuv (muloqot) nuqtasi. Ikkita dastur bir joyda (birgalikda) ishlayotgan vaqtda ularning interfeysi o‘zida uchbu dasturlarning biridagi ikkinchi dastur bilan shug‘ullanadigan barcha xususiyatlarini namoyon qiladi. Dasturning foydalanuvchi interfeysi



foydalanuvchi bilan muloqot qiladigan dasturning barcha jihatlarini o'z ichiga oladi: dastur bilan ishlash uchun zarur bo'lgan sintaksis va semantic qoidalar, dasturga kiritish va chiqarish, dasturning xato va axborot xabarlari. Qobiq dasturlari va ularning har bir utilitlari hamda ularga o'rnatilgan komandalar foydalanuvchi interfeysi hisoblanadi. Standartlashdirish bo'yicha xalqaro tashkilot. ISO-ga qarang

**International  
Organization  
for  
Standardization  
IP**

Internet Protocol. TCP/IP tarmoq qatlami. IP-ma'lumotlarni uzatish qatlami yordamida tarmoq paketlarini boshqarish – tarmoqqa uzatilgan ma'lumotlarni paket (datagram)larga ajratish va qabul qilishda birlashtirishni ta'minlaydigan protocol.

**IP address**

Internet Protokol adresi. Tizim uchun aniq tarmoqdagi ulanishga bog'liq bo'lgan to'rt qismdan iborat adres, Internet protokol (IP) dan foydalaniladi. IP-dan foydalanib bir nechta tarmoqqa ulangan tizim, har bir tarmoq interfeysi uchun alohida IP-adresga ega bo'ladi.

**IPv4**

IP-ning 4-versiyasi. IP va IPv6-ga qarang.

**IPv6**

IP-ning 6-versiyasi. Internet protokolning keying (zamonaviy) avlodi bo'lib, Ipv4-ga ( $2^{32}$  ta) nisbatan ko'proq adreslar ( $2^{128}$  ta) fazosini ta'minlaydi. Bunda internet (tarmoqqa) ulanishi mumkin bo'lgan qurilmalar soni keskin ravishda ko'payib borayotganligi e'tiborga olingan.

IPv6-da o'rnatilgan avtokonfiguratsiyalash mavjud, shuningdek yaxshilangan xavfsizlik, ko'padresli junatmalar va boshqa ko'plab funksiyalar mavjud.

**ISDN**

Integrated Services Digital Network (Integrallashgan xizmatlarning raqamli tarmog'i). Bir juft raqamli yoki standart telefon kabellari orqali tovush, video va ma'lumotlarni sekundiga 64 kilobit tezlik bilan uzatish imkonini beruvchi aloqa standartlari to'plami.

<b>ISO</b>	International Organization for Standardization (Standartlashtirish bo'yicha xalqaro tashkilot). 1946-yilda ta'sis qilingan. Bu tashkilot ko'plab sohalarda, jumladan, kompyuter va kommunikatsiya (aloqa) sohalariga oid standartlar ishlab chiqish uchun mas'ul. Unga 89 ta davlatning standartlashtirish bo'yicha milliy tashkilotlari hamda Amerika milliy standartlashtirish instituti (ANSI) a'zo hisoblanadi.
<b>ISP</b>	Internet service provider (Internet xizmatlari provayderi). Istemolchilarga Internet xizmatlariga kirish huquqini beradi.
<b>kernel</b>	Yadro. Operatsion tizimning bir qismi bo'lib, kompyuter resurslari: xotira, disk sohasi va CPU vaqti, hamda kompyuterda ishlayotgan barcha dasturlarini qamrab oladi. Yadro quyi qatlamdagi apparat interfeyslari (drayverlari)ni o'z ishiga oladi va Linux bajaradigan dasturlar yordamida jarayonlarni boshqaradi. Yadro birinchi bo'lib Linus Tarvalds tomonidan yozilgan Linux tizimining bir qismi hisoblanadi.
<b>LAN</b>	Local area network (mahalliy tarmoq). Mahalli hududda joylashgan (masalan, bitta bino yoki bitta bo'limda) kompyuterlarni birlashtiruvchi tarmoq.
<b>LDAP</b>	Lightweight Directory Access Protocol (Katologlarga kirishning soddalashtirilgan protokoli). Kataloglar xizmatiga on-layn kirishning oddiy protokoli. LDAP – bu katologga kirish protokoli X.500 (DAP - Directory Access Protocol)ning soda muqobili. Undan insonlar haqidagi ma'lumotlarga, tizim foydalanuvchilari, tarmoq qurilmalari, electron pochta kataloglari va tizimlariga kirish uchun foydalanish mumkin. Ba'zi hollarda NIS-ga o'xshash xizmatlarga kirish uchun muqobil sifatida foydalanish mumkin.
<b>log in</b>	Kompyuter tizimiga kirish huquqini olish uchun tizimga kirish so'roviga, ya'ni parolga to'g'ri javob berish
<b>log out</b>	Tizimga kirishda qobiqdan chiqish orqali joriy

	sessiyani yakunlash
<b>MAC address</b>	Media Access Control address (Muhitga kirishni boshqarish adresi). Umumiy tarmoqqa ulangan qurilmaning unikal (takrorlanmas) adresi. Har bir tarmoq adapteri yagona global MAC-adresga ega bo‘ladi. Bu adres DEQQ (doimiy eslab qoluvchi qurilma)da saqlanadi. MAC-adres uzunligi 6 baytdan iborat bo‘lib, $256^6$ (300 trilion atrofida)ta mumkin bo‘lgan adreslarni qo‘llaydi.
<b>MAN</b>	Metropolitan area network (Hududiy tarmoq). Bu tarmoq kompyuterlar va LAN tarmoqlarini birlashtiradigan, bir yoki bir nechta hududlar yoki shaharlar tarmog‘i hisoblanadi.
<b>MDA</b>	Mail delivery agent (Pochtani yetkazish agenti). Pochta tizimining uchta komponentidan biri, qolgan komponentlar MTA va MUA. MDA MTA-dan kiruvchi pochtani qabul qiladi va uni lokal foydalanuvchiga yetkazib beradi.
<b>MIME</b>	Multipurpose Internet Mail Extension (Ko‘p maqsadli Kengaytirilgan Internet Pochtasi). Ilgari, elektron pochtaga biriktirilgan fayllarni ochish uchun uning kengaytmasi (formati)ni oldindan ko‘rib chiqish zarur bo‘lar edi. Hozirgi kunda MIME –ni qo‘llash orqali biriktirilgan faylning kengaytmasiga mos qanday ochilishi yoki ishlashi zarurligi aniqlanadi. MIME turiga misol sifatida image/jpeg –ni olish mumkin: bunda MIME guruhidagi tasvir va MIME – ning turi jpeg. MIME guruhining ko‘plab to‘plari mavjud, jumladan, ilovalar, audio, tasvirlar, xabarlar, matnlar va video ma’lumotlar.
<b>mount</b>	Tizim foydalanuvchilari uchun fayl tizimini ochishga ruxsat beruvchi komanda. Agar fayl tizimi mount qilinmagan (ulanmagan) bo‘lsa, u holda ushbu fayl tizimidagi fayllarni o‘qish va unga yozish mumkin bo‘lmaydi.
<b>MTA</b>	Mail transfer agent (pochtani almashish agenti). Pochta tizimining uchta komponentidan biri. Qolgan ikkita komponenti MDA va MUA. MTA

	foydalanuvchilar va boshqa MTA-lardan xabar qabul qiladi.
<b>MUA</b>	Mail user agent (foydalanuvchi pochta agenti). Pochta tizimining uchta komponentidan biri. Qolgan ikkita komponenti MDA va MTA. MUA – bu oxirgi tugundagi foydalanuvchining elektron pochta dasturi, masalan, KMail, Mutt yoki Outlook.
<b>NAT</b>	Network Address Translation (tarmoq adreslari tarjimoni (akslantiruvchisi)). Lokal tarmoqda ichki tugunlar uchun alohida va tashqi tarmoq uchun alohida IP-adreslar to‘plami qo‘llaniladi. Ichki IP-adreslar to‘plami LAN – xususiy tarmog‘i uchun qo‘llaniladi. Tashqi IP-adreslar to‘plami esa Internet tarmog‘i uchun qo‘llaniladi va odatda ular Internetda unikal adreslar hisoblanadi. NAT ichki IP-adreslarni yopish orqali ba’zi konfidentsiallikni va bir nechta ichki adreslarning Internetga bitta tashqi unikal IP-adres orqali ulanishini ta’minlab beradi.
<b>NFS</b>	Network Filesystem (tarmoq fayl tizimi). Sun Microsystems tomonidan ishlab chiqilgan, UNIX tizimining ko‘pgina ishlab chiqaruvchilari kompyuterlarida mavjud bo‘lgan masofadagi fayl tizimi.
<b>NIC</b>	Network interface card (or controller) (Tarmoq interfeysi kartasi (yoki tekshirgich)). Tarmoqqa fizik ulanishni ta’minlash uchun kompyuterga o‘rnatilgan elektron adapter kartasi
<b>NIS</b>	Network Information Service (tarmoq axborot xizmati). Tizimga bog‘liq bo‘lmagan ma’lumotlarni (masalan, foydalanuvchi nomi va parolini) boshqarish uchun umumiy ma’lumotlar bazasiga asoslangan taqsimlanuvchi xizmati.
<b>open source</b>	Dasturni litsenziyalash va tarqatish usuli bo‘lib, ko‘ngillilar tomonidan ishlab chiqilgan dasturiy ta’minotdan foydalanishni va uni takomillashtirishni qo‘llab-quvvatlashga mo‘ljallangan. Bu usulda tarqatiladigan dasturlarning

dasturlash tilidagi kodini ko‘chirib olish va uni erkin tarzda o‘zgartirish mumkin. Bu “ochiq kiruvchi kod” atamasi hozirgi kunda oldin qo‘llanilgan “erkin dasturi ta’minot” (erkin dasturiy ta’minot fondi [www.fsf.org](http://www.fsf.org) tomonidan ilgari surilgan) atamaga nisbatan keng qo‘llanilmoqda. Lekin tom ma’noda bu usul ham dasturlarni tarqatishga qo‘yiladigan cheklovlardan holi va bepul emas.

**operating system**

Kompyuter resurslarini taqsimlaydigan, jarayonni rejalashtiradigan va foydalanuvchilarga resurslardan foydalanishga yo‘l beruvchi kompyuterni boshqarish dasturi.

Foydalanuvchi, komp’yuter va komp’yuter resurslari o‘rtasida muloqotni tashkil qiluvchi tizimli dasturlar majmuasi.

**option**

Command-line (komandalar satri)ning argumenti bo‘lib, komanda samaradorligini o‘zgartirish uchun qo‘llaniladigan parametrlar. Bu parametrlar tire belgisi bilan boshlanadi va bitta belgidan iborat bo‘ladi (masalan, -h yoki -n). Ba’zi parametrlar guruhlariga birlashtirilgan holda qo‘llaniladi, masalan, -hn. GNU utilitlarida ba’zan ikkita argumentli bitta vazifani bajaradigan to‘liqroq tavsiflangan parametrlar mavjud, bular ikkita tiredan keyin yoziladi (misol uchun, --show-all va --invert-match).

**ordinary file**

Oddiy fayl. Dasturlar, matn yoki foydalanuvchining boshqa ma’lumotlarini saqlovchi fayl.

**PAM**

Linux –ning autentifikatsiya moduli - PAM yoki Linux-Pluggable. Bu model foydalanuvchilarning turli xil ilovalarga qanday autentifikatsiya qilishni tizim administratori aniqlashi uchun qo‘llaniladi.

**passive FTP**

FTP-ning brandmauer orqali ishlash imkoni beradi, bunda klient FTP dasturi o‘rniga ma’lumotlar oqimini ajratib olish va nazorat qilish server tomonidan bajariladi. Shuningdek, PASV FTP deb ham ataladi, sababi u FTP PASV komandasidan

	foydalanadi.
<b>password</b>	Tizimga ruxsat berilmagan foydalanuvchilarning kirishini taqiqlash uchun qo'llaniladigan, erkin holatda foydalanuvchi yoki tizim administratori tomonidan kiritiladigan belgilar ketma-ketligi.
<b>PID</b>	Process identification (jarayon identifikatsiya raqami), odatda raqamli so'z orqali aniqlanadi. Tizimdagi har bir jarayonni boshqasidan ajratish (identifikatsiyalash) uchun Linux OT o'ziga xos (unikal) PID raqamini belgilaydi.
<b>pipe</b>	Bajarilayotgan dasturlar (jarayonlar) o'rtasidagi bog'lanish (ma'lumot almashish) kanali, ya'ni birinchi jarayondagi standart chiqish boshqa (undan keyingi) jarayonning standart kirishiga bog'lanishini ta'minlaydi.
<b>port</b>	Aloqa tizimlaridagi mantiqiy bog'lanish nuqtasi yoki tizimdagi oxirgi tugunga bog'lanish nuqtasi. Transport sathidagi TCP va UDP protokollari Ethernetni qo'llagan holda bitta komp'yuterdagi bitta tarmoq interfeysida turli mantiqiy kanallarni bir biridan ajratish uchun port nomerlaridan foydalanadi.
<b>process</b>	Jarayon. Linux-dagi komanda (utilitlar)ning bajarilishi.
<b>proxy</b>	Tizimning bir bo'lagi sifatida qaralmasada, tizimning faoliyatini amalga oshirish huquqiga ega bo'lgan xizmat. Shuningdek, proxy gateway va proxy server tushunchalari ham mavjud.
<b>proxy server</b>	Foydalanuvchi (klient) lar tez-tez murojaat qiladigan veb-sahifalarni server keshida saqlab turuvchi proxy gateway (proksi shlyuz) ushbu sahifaga navbatdagi murojaatda lokal mavjud bo'lgan resurs sifatida javob qaytaradi (bu tizimning tez ishlashini ta'minlaydi). "proksi-server" va "proksi-shlyuz" atamaları o'zaro o'rin almashadi, shuning uchun ham keshni ishlatish faqat "proksi-server"ga tegishli emas.
<b>RFC</b>	Request for comments (Izoh (komentariya) uchun so'rov). RFC qisman standart hisoblanadi, lekin

barcha internet-standartlar RFC –da yozilgan. Eng ta'sirchan RFC Internetdagi elektron pochtaning standart formati RFC 822 hisoblanadi.

**root directory** Tizimdagi barcha katologlarni ildizi (ajdodi) va barcha fayllarga absolyut yo'l ushbu katolog bilan boshlanadi. Ildiz katologi nomga ega emas va / belgisidan oldin turuvchi alohida katolog hisoblanadi.

**root filesystem** Tizimni bir foydalanuvchi rejimi yoki tiklash rejimida ishga tushirganda kirish mumkin bo'lgan fayl tizimi. Bu fayl tizimi hamma vaqt / orqali ifodalanadi. Bu ildiz fayl tizimini o'rnatish yoki o'chirish mumkin emas. Uning ulanishini o'zgartirish uchun faqat qayta sozlash mumkin.

**root login** Super (root) foydalanuvchining tizimdagi nomi

**root (user)** Super (root) foydalanuvchi uchun boshqa (ikkilamchi) nom

**rotate** O'rin almashtirish. Log fayliga o'xshash fayllar hajmi vaqt o'tishi bilan kattalashib boradi, bunda faylning diskda katta joy egallashiga yo'l qo'ymaslik kerak. Sababi bu log fayllariga yaqin kelajakda qayta murojaat qilishga to'g'ri keladi, qoida sifatida faylning mazmunini (tarkibini) o'chirish tavsiya etilmaydi. O'chirib yubormasdan vaqti vaqti bilan log faylni yangi nom bilan muntazam ravishda saqlab turish va joriy (ishchi) log fayl sifatida yangi bo'sh yaratish mumkin. Keyin ushbu fayllarni seriyali nomlar bilan saqlab, ularga murojaat zarur bo'lganda o'rnini almashtirishingiz mumkin bo'ladi. Misol uchun, xyzlog.4 faylini o'chirganda xyzlog.3→xyzlog.4, xyzlog.2→xyzlog.3, xyzlog.1→xyzlog.2, xyzlog→xyzlog.1 kabi o'rin almashtirishlarni amalga oshirib, yangi xyzlogfile faylni hosil qilish mumkin. Bunday almashtirishlardan keyin xyzlog.4 faylini o'chirish vaqtida o'chirib tashlamoqchi bo'lgan ma'lumotlardan ko'ra yangiroq ma'lumotlar paydo bo'ladi.

**RPC** Remote procedure call (Protosedurlarni masofadan

	chaqirish). Tarmoqda oshkora ishlaydigan protseduralar. Protseduraning o'zi foydalanilayotgan tarmoqqa kirish ruxsatiga javob beradi. RPC bibliotekalari ilovalar uchun tarmoqqa kirishning oshkoraligini ta'minlaydi. RPC TCP/IP yoki UDP/IP-dan yuqori pog'onada ishlaydi.
<b>Samba server</b>	Server Message Block (SMB) protokollarini amalga tatbiq qiladigan bepul dasturlar to'plami. Klient (kompyuter yoki dastur)larga so'rovlar bo'yicha markazlashgan xizmatlarni ta'minlab beruvchi kompyuter (yoki dastur).
<b>shell</b>	Linux-ning tizimli komandalar protsessori (komandalar qobig'i). Asosiy qobiqlaridan biri bu Bourne Again Shell hisoblanadi.
<b>shell script</b>	Shell komandalarini o'z ichiga olgan ASCII fayli. Shuningdek shell dasturi deb ham ataladi.
<b>signal</b>	UNIX tizimi standart kirishlardan tashqari jarayonlarga uzatishi mumkin bo'lgan eng qisqa xabar.
<b>SMB</b>	Server Message Block (Server xabarlarini bloki). 1980-yillar boshida Intel, Microsoft va IBM tomonidan ishlab chiqilgan protokol. SMB - Windows tizimida fayllarni almashish usuli va printerlarga ma'lumotlarni uzatishda klient/server protokoli sifatida taqdim etiladi. Shuningdek, SMB kommunikatsiya kanallari va ketma-ket portlar bilan hamkorlikda ishlaydi, ya'ni nomlangan kanallar va elektron pochta bilan. SMB fayl tizimiga kirish uchun sozlangan masofadan protseduralarni chaqirishga o'xshaydi.
<b>SMP</b>	Symmetric multiprocessing (Simmetrik ko'pprotsessorli qayta ishlash). Yuqori o'tkazuvchan kanal orqali bog'langan ikki va undan ortiq bir xil protsessorlar bitta operatsion tizim orqali boshqariladi, bunda har bir protsessorga kiritish-chiqarish qurilmalaridan bir xil kirish huquqi berilgan. Bu protsessorlar bir ixl darajada qayta ishlaydi, shuning uchun ham operatsion tizim talabi asosida amaliy dasturlar



ixtiyoriy bittasida yoki o‘zaro hamkorlikda barcha protsessorlarni ishlatishi mumkin.

## **SMTP**

Simple Mail Transfer Protocol (Oddiy pochta uzatish protokoli). Tarmoqda elektron pochta jo‘natmalarini almashish uchun foydalaniladigan protokol. Bu serverdan serverga (server-to-server) protokoli hisoblanadi, shuning uchun ham xabarlarga kirish uchun boshqa protokollar qo‘llaniladi. SMTP protokoli odatda xabarlarni uzatish tizimi boshqaruvida sendmail yoki exim4 kabi fon rejimida ishlaydi

## **spam**

Ish jarayoniga bog‘liq bo‘lmagan yoki bir yoki bir nechta xabarlar guruhiga mos keluvchi tarmoq xatoliklaridan kelib chiqadigan pochta xabarlari. Bundan tashqari, katta hajmdagi nomaqbul elektron pochta jo‘natmalarini yuborish. Bundan elektron jo‘natmalar odatda mahsulotlar yoki xizmatlarni reklama qilish uchun qo‘llaniladi. Spamlarning yana bir umumiy maqsadi xakkerlik hujumlarini tashkil qilish hisoblanadi.

## **standart input**

Dasturlar kirish ma‘lumotlarini olishi mumkin bo‘lgan fayl. Agar fayl ko‘rsatilmagan bo‘lsa, bu kirish klaviaturaga yo‘naltiriladi (ya‘ni, klaviaturani tavsiflovchi qurilma faylidan ma‘lumotlarni oladi).

## **standart output**

Dastur chiqish ma‘lumotlarini uzatishi mumkin bo‘lgan fayl. Agar fayl ko‘rsatilmagan bo‘lsa, dastur bu chiqish ma‘lumotlarini ekranga uzatadi (ya‘ni, ekranni tavsiflovchi qurilma fayliga yuboradi)

## **startup file**

Qobiqqa kirishda ishga tushiriladigan fayl. Bourne Again Shell va Z Shell qobiqlari .profile-ni ishga tushiradi, C Shell esa .login-ni ishga tushiradi. C Shell yangi C Shell qayta chariliganda .cshrc-ni qayta ishga tushiradi. Z Shell ham shunga o‘xshash nomi ENV o‘zgaruvchisi bilan identifikatsiya qilinadigan faylni ishga tushiradi.

## **Superuser**

Root huquqi bilan ishlovchi foydalanuvchi. Bu foydalanuvchi komp’yuterning ixtiyoriy resursiga

kirish imkoniyatiga ega bo'ladi. Tizim administratori yangi foydalanuvchilarni yaratish, parollarni almashtirish yoki boshqa boshqarish funksiyalarini amalga oshirishi uchun Superuser holatida bo'lishi shart. Odatda Superuser foydalanuvchisining nomi root bo'ladi. Shuningdek, root foydalanuvchisi yoki oddiygina root deb ham ataladi.

**system administrator**

Tizim administratori. Tizimga xizmat ko'rsatishga mas'ul bo'lgan shaxs. Tizim administratori tizimga root yoki sudo-dan foydalanib kirishi mumkin. Superuser-ga qarang.

**System V TCP**

UNIX tizimining ikkita asosiy versiyalaridan biri Transmission Control Protocol (Yetkazishni nazorat qilish protokoli). Internetda qo'llaniladigan eng ko'p tarqalgan transport sathi protokoli. Ushbu bog'lanishga yo'naltirilgan protokol IP protokoli ustiga qurilgan va TCP/IP (TCP orqali IP) juftligida qo'llaniladi. TCP oqimlar ketma-ketligini boshqarishda ishonchli aloqa o'rnatadi va jarayonlar o'rtasida to'liq dupleksli bog'lanishni ta'minlaydi. UDP esa, bog'lanishni o'rnatmasdan IP protokolidan yuqorida ishlovchi protokol hisoblanadi.

**terminal**

Ishchi stantsiyalardan intellekti yetishmovchiligi bilan farq qiluvchi qurilmalar. Terminal komp'yuterga Linux boshqaruvchi asosida ulanadi. Ishchi stantsiyalar Linux tizimi bilan ishlaydi.

**TTL**

Yashash vaqti.

1. Barcha DNS yozuvlarida ko'rsatiladi, odatda bir haftadan ko'p bo'lmagan davr qayd qilinadi. Bu vaqt yozuvlarning yashash davri deb ataladi. DNS-server yoki ilova dasturlari bu yozuvlarni keshda saqlaganda, ular TTL qiymatini kamaytiradi va qiymat nolga tenglashganda yozuvni keshdan o'chiradi. DNS-server keshlangan yozuvni bu yozuv nechta server orqali o'tishiga bog'liq bo'lmagan holda TTL-ga tegishlilikni kafolatlovchi joriy (kamaytirilgan) TTL bilan boshqa serverlarga

yuboradi.

2. IP sarlavhasidagi paketni qaytarish yoki tashlab yuborishdan oldin nechta paketlar to'plamini o'tkazish kerakligi ko'rsatiladigan maydon.

<b>UDP</b>	User Datagram Protocol (Foydalanuvchining datagramm protokoli). Oddiy, lekin ishonchsiz xizmatlarni ta'minlaydigan Internet standartidagi transport protokol. UDP protokoli TCP kabi aloqa o'rnatmaydigan IP-dan yuqori sathdagi protokol.
<b>UID</b>	User ID. Passwd (parollar) bazasidagi foydalanuvchi nomi bilan bog'liq bo'lgan raqam.
<b>URI</b>	Universal Resource Identifier (Universal resurs identifikatori (nomi)). Qisqa satrli ob'ektlarga (odatda Internetdagi) tegishli bo'lgan barcha nomlar va adreslarning umumiy to'plami. URI eng ko'p tarqalgan ko'rinishi URL hisoblanadi.
<b>URL</b>	Uniform (was Universal) Resource Locator (Yagona (universal) resurslar joylashuvi). Internetdagi ob'ekt, odatda veb-sahifalarni aniqlashning standart usuli. URL-lar URI-larning quyi sathi hisoblanadi.
<b>VPN</b>	Virtual private network (virtual xususiy tarmoq). Internetga o'xshash jamoaviy tarmoq. VPN biror kompaniyaga tegishli bo'lgan xususiy tarmoq bo'lib, ma'lumotlarni almashishda maxfiylikni ta'minlash uchun shifrlashdan foydalanadi.
<b>WAN</b>	Wide area network (butun jahon tarmog'i). Keng geografik maydondagi LAN va MAN tarmoqlarini birlashtiruvchi tarmoq.
<b>working directory</b>	Foydalanuvchi tez-tez murojaat qiladigan katalog. Joriy vaqtda ishlab turgan katalog.
<b>WYSIWYG</b>	What You See Is What You Get (nimani ko'rsang, shuni olasan). Matn protsessorlaridagi pechatga chiqarish ekraniga o'xshash grafik ilova. Odatda veb-sahifalarni ishlab chiqishda qo'llaniladigan usul.

# MUNDARIJA

<b>KIRISH.....</b>	<b>3</b>
<b>1-BOB. SHELL SCRIPT ASOSLARI .....</b>	<b>4</b>
1.1-§. SHELL SCRIPT – ASOSIY TUSHUNCHALAR.....	4
1.2-§. BOURNE SHELL ILOVASINI YUKLASH .....	5
1.3-§. O‘ZGARUVCHILAR .....	5
1.4-§. MASSIV O‘ZGARUVCHILARI.....	7
1.5-§. TASHQI O‘ZGARUVCHILAR .....	8
1.6-§. FOYDALANUVCHI BILAN MULOQOT .....	10
1.7-§. O‘RNIGA QO‘YISH. ....	11
1.8-§. ARIFMETIK IFODALAR .....	12
1.9-§. IF SHARTLI OPERATORI .....	13
1.10-§. SHARTNI TEST DASTURI YORDAMIDA TEKSHIRISH.....	14
1.11-§. O‘RNATILGAN OPERATORLAR && VA    LARNI QO‘LLASH .....	15
1.12-§. CASE OPERATORI .....	16
1.13-§. FOR OPERATORI.....	17
1.14-§. TASHQI FAYLDAN MA‘LUMOT OLISH.....	19
1.15-§. WHILE OPERATORI .....	20
1.16-§. SELECT OPERATORI .....	21
1.17-§. NUQTA OPERATORI .....	22
1.18-§. MAXSUS O‘ZGARUVCHILAR .....	24
1.19-§. BOSHQA MAXSUS O‘ZGARUVCHILAR .....	25
1.20-§. GETOPTS DASTURINI QO‘LLASH .....	26
1.21-§. TRAP OPERATORI .....	27
<b>2-BOB. LINUXNING INITSIALIZATSIYA (ISHGA TUSHIRISH) TIZIMI... 29</b>	<b>29</b>
2.1-§. LINUX ISHGA TUSHIRISH TIZIMLARI.....	29
2.2-§. INIT DASTURI.....	30
2.3-§. /ETC/INITTAB FAYLINING FORMATI .....	32
2.4-§. BSD INITSIALIZATSIYA TIZIMI .....	36
2.5-§. SYSTEMV INITSIALIZATSIYA TIZIMI.....	56
2.6-§. CHKCONFIG DASTURI .....	61
2.7-§. XIZMATLARNI QO‘LDA YUKLASH VA TO‘XTATISH.....	62
2.8-§. /ETC/SYSCONFIG KATOLOGIDAGI KONFIGURATSIYA FAYLLARI.....	63

2.9-§. ROOT FOYDALANUVCHISINING PAROLINI TIKLASH .....	63
<b>3-BOB. LINUXDA QURILMALAR BILAN ISHLASH .....</b>	<b>67</b>
3.1-§. LINUXDA QURILMALAR BILAN ISHLASH .....	67
3.2-§. LINUX YADRO MODULINI YUKLASH .....	68
3.3-§. MODPROBE DASTURINING KONFIGURATSIYASI .....	69
3.4-§. LDCONFIG VA LDD DASTURLARI .....	72
3.5-§. IDE KONTROLLERNI SOZLASH.....	73
3.6-§. TIZIMNING DASTLABKI YUKLANISHIDA QATTIQ DISK PARAMETRLARINI SOZLASH .....	76
3.7-§. FOYDALANUVCHI LOGININI SOZLASH .....	79
3.8-§. LOGIN DASTURINING KONFIGURATSIYASI.....	80
3.9-§. /ETS/LOGIN.ACCESS VA /ETC/SECURE/ACCESS-CONF FAYLLARI .....	83
3.10-§. RAM MODULI VA UNING KONFIGURATSIYA FAYLI. ....	83
3.11-§. DISK MAYDONIDAN FOYDALANISHNI CHEKLASH. ....	86
3.12-§. QUOTACHECK DASTURI .....	87
3.13-§. QUNTAON DASTURI .....	87
3.14-§. QUOTAOFF DASTURI .....	88
<b>4-BOB. LOG FAYLLARI BILAN ISHLASH .....</b>	<b>91</b>
4.1-§. SYSLOG TIZIMI .....	91
4.2-§. SYSLOGD DASTURINING KONFIGURATSIYA FAYLI .....	91
4.3-§. XABARLARNING MUHIMLIK DARAJASI VA VOSITALARI.....	93
4.4-§. SYSLOG TIZIMI ISHINING BA’ZI XUSUSIYATLARI .....	97
4.5-§. TIZIM FOYDALANUVCHILARI HISOBI.....	99
4.6-§. XAVFSIZLIK MASALALARI .....	100
4.7-§. DASTURLARNING JADVAL BO’YICHA BAJARILISHI .....	103
4.8-§. AT VA ATD DASTURLARI .....	108
4.9-§. NAVBATNING BAJARILISHI VAQTIDA TOPSHIRIQLARNI KUZATISH .....	108
4.10-§. LOG FAYLLARINI ALMASHTIRISH (ROTATSIYA) .....	110
4.11-§. TARMOQ SUPERSERVERI .....	114
<b>5-BOB. DNS SERVERNI SOZLASH.....</b>	<b>120</b>
5.1-§. DOMAIN NAME SYSTEM .....	120
5.2-§. DNS SERVERNING JAVOBGARLIK HUDUDI (SOHASI).....	121

5.3-§. DNS-NING ISHLASH PRINSIPLARI .....	122
5.4-§. KESHLOVCHI DNS SERVER .....	123
5.5-§. BIND DNS SERVERI .....	126
5.6-§. BIND SERVERINING KONFIGURATSIYA FAYLLARI.....	127
5.7-§. HUDUDLARNI TAVSIFLASH FAYLI .....	131
5.8-§. RESURLAR HAQIDA YOZUV .....	132
5.9-§. NAMED.CONF FAYLINING PARAMETRLARI .....	138
5.10-§. HUDUDNI TAVSIFLASH FAYLI.....	139
5.11-§. DOMEN UCHUN MAS'UL DNS SERVERLARNI ANIQLASH.....	140
5.12-§. SINTAKSIS XATOLARNI TEKSHIRISH UCHUN UTILITLAR .....	141
5.13-§. DNS SERVERNI BOSHQARISH .....	142
5.14-§. DNS SERVERNI YUKLASH .....	143
5.15-§. DNS KLIENI SOZLAMALARI.....	144
5.16-§. SLAVE HUDUDINI QO'LLASH SOZLAMALARI .....	146
5.17-§. DOMENNI BOSHQARISH HUQUQINI TAYINLASH.....	146
5.18-§. ICHKI DOMENLARDAN FOYDALANISH .....	147
5.19-§. DNS SERVERNI TESTLASH VA TUZATISH (OTLADKA QILISH) .....	151
5.20-§. TESKARI AKSLANTIRISH HUDUDI .....	153
5.21-§. KESHLOVCHI SERVER.....	154
5.22-§. XAVFSIZLIK MASALALARI .....	156
5.23-§. DNS SERVERLARDA MA'LUMOT ALMASHISH .....	157
5.24-§. DNS SERVERNI ODDIY FOYDALANUVCHI HUQUQIDA ISHGA TUSHIRISH .....	159
<b>6-BOB. ELEKTRON POCHTA.....</b>	<b>162</b>
6.1-§. ELEKTRON POCHTA KOMPONENTLARI.....	162
6.2-§. SMTP PROTOKOLI .....	164
6.3-§. SENDMAIL .....	167
6.4-§. SENDMAIL-NING ISH REJIMI.....	171
6.5-§. FOYDALANUVCHINING POCHTA QUTISI .....	172
6.6-§. YUBORISHDA NAVBAT BILAN ISHLASH .....	173
6.7-§. LOG TIZIMI.....	175
6.8-§. ODDIY POCHTA SERVERINI SOZLASH .....	177
6.9-§. KONFIGURATSIYA FAYLLARINI HOSIL QILISH.....	178
6.10-§. POCHTA SERVERLARINING VIRTUAL XOSTINGI.....	181

6.11-§. DOIMIY INTERNET-GA ULANGAN TASHKILOT UCHUN POCHTA SERVERINING VIRTUAL XOSTINGINI TASHKIL ETISH .....	183
6.12-§. SENDMAIL-NING QO‘SHIMCHA KONFIGURATSIYASI.....	185
6.13-§. POCHTA PSEVDONIM(NOMLARI)LARI .....	192
6.14-§. ELEKTRON POCHTA ADRESLARI RO‘YXATINI FAYLDAN YUKLASH .....	193
<b>7-BOB. SQUID PROXY SERVERI .....</b>	<b>196</b>
7.1-§. KIRUVCHI KODLARDAN DASTURNI YIG‘ISH.....	196
7.2-§. SQUID.CONF KONFIGURATSIYA FAYLI.....	197
7.3-§. TARMOQQA ULANISHGA TA‘SIR QILUVCHI PARAMETRLAR.....	198
7.4-§. PROKSI SERVERLAR IERARXIYASINI TASHKIL ETISH PARAMETRLARI .....	199
7.5-§. PROKSI SERVER KESHI O‘LCHAMINI ANIQLASH PARAMETRLARI .....	200
7.6-§. LOG FAYLLARI VA KESHNING JOYLASHISH O‘RNINI ANIQLOVCHI PARAMETRLAR .....	201
7.7-§. QO‘SHIMCHA DASTURLARNING ISHLASHIGA TA‘SIR ETUVCHI PARAMETRLAR .....	202
7.8-§. KIRISHNI NAZORAT QILISH RO‘YXATI (ACL).....	203
7.9-§. KIRISHNI CHEKLASH, HTTP_ACCESS PARAMETRI .....	206
7.10-§. BOSHQARUV (ADMINISTRATIV) PARAMETRLARI .....	207
7.11-§. PROXY SERVERNI BOSHQARISH.....	208
7.12-§. AUTENTIFIKATSIYANI SOZLASH .....	209
7.13-§. KANAL KENGLIGINI CHEKLASH. ....	211
7.14-§. SQUID PROXY SERVERINING LOG FAYLLARI BILAN ISHLASH.....	213
<b>8-BOB. LINUX-DA PAKETLARNI FILTRLASH (FIREWALL) .....</b>	<b>215</b>
8.1-§. LUNIX-DA PAKETLARNI FILTRLASH (FIREWALL).....	215
8.2-§. ZANJIRLAR VA JADVALLARDAN O‘TISH TARTIBI .....	217
8.3-§. IPTABLES DASTURI .....	219
8.4-§. PAKETLARNI AJRATISH MEZONLARI. ....	221
8.5-§. OSHKOR MEZONLAR .....	226
8.6-§. AMALLAR VA O‘TISHLAR. ....	230
8.7-§. NAT AKSLANTIRISHLAR .....	234
8.8-§. NETFILTER-NI BOSHQARISH .....	237
<b>9-BOB. NFS -TARMOQ FAYL TIZIMI. APACHE VA FTP SERVERLARI</b>	<b>244</b>

9.1-§. NFS – TARMOQ FAYL TIZIMI.....	244
9.2-§. NFS SERVERI.....	246
9.3-§. NFS-KLIENTI .....	248
9.4-§. APACHE WEB SERVERI.....	249
9.5-§. APACHE WEB-SERVERINING PARAMETRLARI .....	250
9.6-§. RESURSLARGA KIRISHNI NAZORAT QILISH .....	253
9.7-§. VIRTUAL XOSTING .....	255
9.8-§. FTP-SERVERI.....	256
9.9-§. FTP-SERVERINI ISHGA TUSHIRISH .....	259
<b>10-BOB. LABORATORIYA MASHG‘ULOTLARI .....</b>	<b>260</b>
1-LABORATORIYA ISHI. SLACKWARE LINUX-DA TARMOQNI SOZLASH .....	260
2-LABORATORIYA ISHI. ROOT FOYDALANUVCHISINING PAROLINI O‘ZGARTIRISH .....	262
3-LABORATORIYA ISHI. IDE KONTROLLERNI SOZLASH.....	263
4- LABORATORIYA ISHI. DISK MAYDONIDAN FOYDALANISHNI CHEKLASH SOZLAMALARI .....	265
5-LABORATORIYA ISHI. SYSLOG TIZIMINI SOZLASH.....	267
6-LABORATORIYA ISHI. AUTH.NOTICE FAYLINI NAZORAT QILISH UCHUN SKRIPT YARATISH VA QO‘LLASH .....	269
7-LABORATORIYA ISHI. CRON-DAN FOYDALANISH. ....	270
8-LABORATORIYA ISHI. AT TIZIMINI QO‘LLASH.....	271
9-LABORATORIYA ISHI. LOGROTATE DASTURINI QO‘LLASH. ....	272
10-LABORATORIYA ISHI. INETD SUPERSERVERINI QO‘LLASH .....	273
11-LABORATORIYA ISHI. TINGLOVCHILAR KOMPYUTERIDA DNS SERVERNI SOZLASH .....	275
12-LABORATORIYA ISHI. SLAVE HUDUDNI SOZLASH .....	277
13-LABORATORIYA ISHI. TESKARI AKSLANTIRISH HUDUDINI SOZLASH .....	278
14-LABORATORIYA ISHI. DNS-NI ODDIY FOYDALANUVCHI HUQUQI BILAN ISHLATISH.....	280
15-LABORATORIYA ISHI. ODDIY POCHTA SERVERINI SOZLASH. ....	281
16-LABORATORIYA ISHI. VIRTUSERTABLE VOSITASINI QO‘LLASH.....	284
17-LABORATORIYA ISHI. KIRISH KODIDAN DASTUR YIG‘ISH.....	285
18-LABORATORIYA ISHI. SQUID PROKSI SERVERINI SOZLASH VA ISHGA TUSHIRISH .....	286



19-LABORATORIYA ISHI. FOYDALANUVCHILAR AUTENTIFIKATSIYASINI SOZLASH .....	288
20-LABORATORIYA ISHI. KANAL KENGLIGINI CHEKLASH SOZLASH.....	290
21-LABORATORIYA ISHI. LIMIT MEZONIDAN FOYDALANISH. ....	291
22-LABORATORIYA ISHI. LOG AMALIDAN FOYDALANISH. ....	293
23-LABORATORIYA ISHI. NAT AKSLANTIRISHNI QO‘LLASH .....	294
24-LABORATORIYA ISHI. NFS-DAN FOYDALANISH.....	295
25-LABORATORIYA ISHI. VIRTUAL XOSTINGNI SOZLASH. ....	297
26-LABORATORIYA ISHI. FTP-SERVERNI SOZLASH. ....	299
<b>FOYDALANILGAN ADABIYOTLAR .....</b>	<b>301</b>
<b>GLOSSARIY .....</b>	<b>302</b>

I.M. BOYNAZAROV

# LINUX OPERATSION TIZIMI

(O'quv qo'llanma)

Toshkent – «Nihol print» OK – 2021

Muharrir: Q.Matqurbonov  
Tex. muharrir: A.Tog'ayev  
Musavvir: B.Esanov  
Musahhiha: G.Tog'ayeva  
Kompyuterda  
sahifalovchi: B.Berdimurodov

Nashr.lits. AI №176. 11.06.11.

Bosishga ruxsat etildi: 12.08.2020. Bichimi 60x841 /16.  
Shartli bosma tabog'i 20,25. Nashr bosma tabog'i 20,0.  
Adadi 100. Buyurtma № 20.